The Dissertation Committee for Shan Yang
certifies that this is the approved version of the following dissertation:

A Shape Hessian-based Analysis of Roughness Effects on Fluid Flows

Committee:

Omar Ghattas, Supervisor

Robert Moser, Supervisor

Todd Arbogast

David Goldstein

Lexing Ying

A Shape Hessian-based Analysis of Roughness Effects on Fluid Flows

by

Shan Yang, B.S.; M.S.C.A.M.

DISSERTATION

Presented to the Faculty of the Graduate School of
The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN
August 2011

Dedicated to my parents.

Acknowledgments

Through this dissertation work, I received a lot of help from multitudes of people and I wish to give my sincere thanks to all those who helped me.

First I want to thank my advisors, Dr. Omar Ghattas, Dr. Robert Moser, Dr. Georg Stadler. I feel fortunate to work with them for the past years. Omar always impressed me with his broad knowledge and I learned a lot from him; Bob brought me into the field of turbulence and taught me how to analyze from the engineering point of view; I could not have come such a long way to achieve my current level of understanding in my research area without the long hours of discussion with Georg and he also gave me a lot help in English writing.

My thanks go to my other committee members, Dr. Todd Arbogast, Dr. David Goldstein and Dr. Lexing Ying. They not only spared their precious time reading this dissertation, but also provided a lot of valuable insight comments.

Working with Dr. Ghattas's group (CCGO) and Dr. Moser's group, I was lucky to get numerous help from other members in the two groups. I want to thank them for all the discussions and comments. Especially, my thanks go to Pearl Flath for her help in the proof-reading of this thesis, Henry Chang for his suggestions and encouragement. I am grateful to the other CAM students. Thank them for their friendship, enthusiasm, and numerous help, especially to Jennifer Worthen, Andrea Hawkins and Ju Liu. I appreciate all the help that the ICES staff has provided to me, especially to Sue Rodriguez, Robert Hoelscher, Fatima Bridgewater and Stephanie Rodriguez.

Most importantly, this work would not have been possible without the love, patience and support of my family. I owe too much to my parents for their years of efforts of bringing me up and to my sister for her love and support. Words are not enough to express my gratitude towards them.

A Shape Hessian-based Analysis of Roughness Effects on Fluid Flows

Publication No. ____________________

Shan Yang, Ph.D.
The University of Texas at Austin, 2011

Supervisors: Omar Ghattas
Robert Moser

The flow of fluids over solid surfaces is an integral part of many technologies, and the analysis of such flows is important to the design and operation of these technologies. Solid surfaces, however, are generally rough at some scale, and analyzing the effects of such roughness on fluid flows represents a significant challenge. There are two fluid flow situations in which roughness is particularly important, because the fluid shear layers they create can be very thin, of order the height of the roughness. These are very high Reynolds number turbulent wall-bounded flows (the viscous wall layer is very thin), and very low Reynolds number lubrication flows (the lubrication layer between moving surfaces is very thin). Analysis in both of these flow domains has long accounted for roughness through empirical adjustments to the smooth-wall analysis, with empirical parameters describing the fluid dynamic roughness effects. The ability to determine these effects from a topographic description of the roughness is limited (lubrication) or non-existent (turbulence). The commonly used parameter, the *equivalent sand grain roughness*, can be determined in terms of the change in the rate of viscous energy dissipation caused by the roughness and is generally obtained by measuring the effects on a fluid flow. However, determining fluid dynamic effects from roughness characteristics is critical to effective engineering analysis.

Characterization of this mapping from roughness topography to fluid dynamic impact is the main topic of the dissertation. Using the mathematical tools of shape calculus, we construct this mapping by defining the roughness functional and derive its first- and second- order shape derivatives, i.e., the derivatives of the roughness functional with respect to the roughness topography. The results of the shape gradient and complete spectrum of the shape Hessian are presented for the low Reynolds number lubrication flows. Flow predictions based on this derivative information is shown to be very accurate for small roughness. However, for the study of high Reynolds number turbulent flows, the direct extension of the current approach fails due to the chaotic nature of turbulent flows. Challenges and possible approaches are discussed for the turbulence problem as well as a model problem, the sensitivity analysis of the Lorenz system.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The flow of fluids over solid surfaces is a common feature of engineered systems. However, the analysis and control of such fluid flows is complicated by the fact that the surfaces involved are generally rough at some scale. Surface topography can have a significant effect on the rate of momentum transfer to the surface (drag). Generally, the magnitude of the roughness effect on a fluid flow varies with the height of the roughness scaled by the thickness of the fluid shear layer in which it exists. In engineered systems, for which surfaces are made to be relatively smooth, these roughness effects are commonly important in two situations: in lubrication applications, where a thin viscous fluid film flows between sliding surfaces; and in turbulent flows, where the turbulence acts to produce a thin viscously-dominated layer very close to the wall (this is the so-called viscous sublayer). In engineering analysis, the effects of roughness have long been accounted for through the use of empirical adjustments to smooth-wall relations. It is currently not possible to determine these effects directly from a description of the roughness topography. However, predicting and understanding how the characteristics of a roughness affect fluid dynamics is important for effective engineering design and analysis.

1.1 Roughness in Lubrication Flows

In lubrication applications the primary analytic tool is the Reynolds lubrication equation, which for simplicity is shown here for steady flow in one dimension (a two-dimensional narrow gap):

$$\frac{1}{12\mu}\frac{\partial}{\partial x}\left(h^3\frac{\partial p}{\partial x}\right)=\frac{U}{2}\frac{\partial h}{\partial x}, \tag{1.1}$$

where U is the velocity of one of the bearing surfaces (the other is fixed), h is the spacing between the surfaces and p is the pressure. The 1-D Reynolds equation is derived by integrating the two-dimensional Stokes flow across the gap between the two moving surfaces, under the assumption that the gap width varies in x over scales $L \gg h$, and that the surfaces are smooth. A common approach to treating roughness in this context is to introduce flow factors $\phi_1(x)$ and $\phi_2(x)$ [57, 58] into the Reynolds equation as follows:

$$\frac{1}{12\mu}\frac{\partial}{\partial x}\left(\phi_1 h^3\frac{\partial p}{\partial x}\right)=\frac{U}{2}\frac{\partial(\phi_2 h)}{\partial x}. \tag{1.2}$$

Since the introduction of roughness flow factors in 1978, there have been a number of approaches proposed to determine them for given roughness topographies. These have included analysis of stochastic models [16, 74, 75], analysis of specified deterministic roughness topographies [43, 46] and asymptotic homogenization analysis [4, 5, 7, 34, 37, 60, 61]. The homogenization analysis has lead to the development of bounds on the roughness effects, and thus the flow factors [6, 49]. While the homogenization approach appears to provide the best opportunity to systematically account for roughness effects, the work cited above is unfortunately limited by the use of the Reynolds equations (1.1) to describe the details of the flow over the roughness topography. This is an issue, because the Reynolds equation is an asymptotic approximation valid for geometric variations that occur over distances large compared to the lubrication film thickness. Roughness need not satisfy this condition.

A more general approach involves asymptotically homogenizing the Stokes equations for flow over a rough surface similar to the homogenization used by [1, 8]. In this case, a rough surface version of the Reynolds equations is obtained directly. The analysis is most easily done using variational forms of the Stokes and Reynolds equations. For example, consider a two-dimensional lubrication film between rough surfaces (a domain Ω), in which the roughness topography varies in x over a characteristic length scale h_0 that also characterizes the film thickness. There are also variations in film thickness that occur over large length scales $L \gg h_0$. This leads to a classical asymptotic (in $\varepsilon = h_0/L$) multiscale representation of the Stokes flow. The resulting large-scale variational statement of the Stokes continuity equation is then to find $P(x)$ (the large-scale pressure does not vary in the cross-flow direction) that minimizes:

$$\int_\Omega \frac{\partial P\overline{u}}{\partial x}\, dv - \int_x U\overline{\tau_w}\, dx + \mu \int_\Omega \overline{\nabla\mathbf{u} : \nabla\mathbf{u}}\, dv. \tag{1.3}$$

In this expression, u is a solution to a small-scale Stokes flow problem in which the two surfaces are rough, but are on average parallel, with one wall moving at velocity U and driven by an average pressure gradient (a pressure driven Couette flow). The over-line signifies the horizontal average in the small-scale problem. In the second term, $\overline{\tau_w}$ is the wall shear stress on the moving surface in the small-scale problem. The integrand in the third term is the horizontally averaged rate of kinetic energy dissipation. Part of the analogous term from the variational statement of (1.2) is

$$\frac{1}{12\mu}\int_x \phi_1 h^3 \left(\frac{\partial P}{\partial x}\right)^2 dx. \tag{1.4}$$

Further analysis indicates that to be consistent, ϕ_1 must simply be the ratio of the y-integrated energy dissipation rate due to the pressure-driven component of the small-scale problem with and without roughness. Similarly, the flow-factor ϕ_2 is related to dissipation associated with the shear driven component of the flow.

In our study, instead of treating roughness in the context of the Reynolds equations, we want to determine the average dissipation rate in Stokes flows on rough surfaces. So we would like to find a functional mapping the roughness topography to the dissipation.

1.2 Rough-wall Turbulence

Roughness can have a very large effect in turbulent flows because the roughness height can be of the same order as the thickness of the near-wall viscous layer. In turbulence [35], roughness is generally parametrized (e.g. equivalent sand-grain roughness height k_s) through its effect on the flow, rather than the surface topography of the roughness. Ideally, given an appropriate characterization of the roughness topography, one would like to simply determine the fluid dynamic effects. This mapping from surface topography to fluid dynamic parameters is currently not available.

In high Reynolds number wall-bounded turbulent shear flows over smooth surfaces, the mean velocity is characterized by two length scales: an inner viscous length scale given by ν/u_τ, where ν is the kinematic viscosity and $u_\tau = \sqrt{\tau_w/\rho}$ is the friction velocity (τ_w is the mean wall shear stress); and an outer length scale δ characterizing the thickness of the turbulent layer (e.g. pipe radius or boundary layer thickness). In the overlap region (asymptotic matching region) between the inner and outer scaling layers, the mean velocity takes the form [70]:

$$U^+ = \kappa^{-1} \log y^+ + A, \tag{1.5}$$

where $U^+ = U/u_\tau$ is the scaled mean velocity and $y^+ = yu_\tau/\nu$ is the scaled height above the smooth surface, $\kappa \approx 0.4$ is the Karman constant, which has been thought to be universal, and A is a constant determined by the details of the inner layer (from experiments $A \approx 5$). This log-layer relation is commonly considered to be valid in a range $y^+ > 50$ and $y/\delta < 0.15$, though there is some controversy about these values (see e.g. [36]), and the universality of κ [54].

In his experiments in pipes with roughness created from graded close-packed sand, Nikuradse [55] found, as expected, that the inner viscous length scale was replaced by the roughness scale, k_s, which he took to be the size of his graded sand grains. The mean velocity in the overlap region was then found to be

$$U^+ = \kappa^{-1} \log y/k_s + A_k, \tag{1.6}$$

where the constant $A_k \approx 8.5$ is different because of the different inner layer scaling. This is now considered to be the defining relation for the equivalent or effective sand grand roughness height. For any rough surface, this ratio is determined by fitting the mean velocity profile to (1.6) in the overlap region. In this context, k_s is simply a convenient way to characterize the change in the drag force (wall shear stress), since from the above, it is a monotonically increasing function of k_s [35].

If k is a geometric measure of the height of the roughness (e.g. rms (root-mean-square) or maximum roughness height), when $k^+ \gg 1$, so that viscosity is negligible, and Reynolds number is very large, so that $k/\delta \ll 1$, dimensional analysis requires that the ratio k_s/k depends only on the roughness geometry. The usual estimate of when this asymptotic behavior is obtained is $k^+ > 40$ and $\delta/k > 50$, where k is here interpreted as the maximum height [35]. This high Reynolds number fully rough condition can be achieved only if the friction Reynolds number of the flow is sufficiently large ($Re_\tau > 2000$), which makes computational studies in this regime challenging.

There have been many efforts over the years to correlate k_s/k with various geometric roughness parameters [12, 63, 66, 73], with some success for geometrically simple, easily described roughness. For irregular roughness as might result from machining processes or surface damage, such attempts have been less successful [71]. What we really want to know in this case is the functional mapping the surface topography to k_s/k, or equivalently, from surface topography with a given k to the surface drag (wall stress).

Many experimental and computational studies have been undertaken to characterize the effects of roughness by investigating flow responses to idealized deterministic roughness (see [13] and references therein). As noted by Schlichting [63], deterministic roughness avoids the lack of repeatability inherent in randomly prepared rough surfaces, such as the sand-grain roughness used by Nikuradse [55]. While such deterministic roughness studies have been of great value, they have inherent shortcomings. In such studies, what one essentially does is a single (or small number of) roughness functional evaluation(s). One is unable to determine how much larger the observed roughness effects can be, nor does one determine what features of the roughness topography are primarily responsible for these effects.

Here, we propose to systematically explore a roughness functional describing the total drag by determining shape gradients and shape Hessians of the functional, and to use it to explore the sensitivity of drag to changes in roughness topographies. By finding roughness topographies that yield particularly large or small drag, and by determining the changes to such topographies to which the drag are most sensitive, we will be able to determine what is important about a roughness topography in determining its effect.

1.3 Shape derivatives approach

The main goal of this dissertation is to explore the functional described above to study the influence of roughness. The functional, to which we refer as the *roughness functional* in the remainder of this dissertation, maps roughness topography to the dissipation rate, which is related to drag through a global energy balance (see Chapter 3). It is given by

$$\mathscr{D}(\Omega) = \frac{\nu}{4} \int_{\Omega} \langle (\nabla \mathbf{u} + \nabla \mathbf{u}^T) : (\nabla \mathbf{u} + \nabla \mathbf{u}^T) \rangle \, d\mathbf{x}, \tag{1.7}$$

where $\Omega \subset \mathbb{R}^N (N = 2, 3)$ is a domain with a rough boundary surface, ν denotes the kinematic viscosity, and $\mathbf{u}$ is the velocity field that satisfies the incompressible Navier-Stokes equations,

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} - \nabla \cdot \left(-Ip + \nu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)\right) + \mathbf{u} \cdot \nabla \mathbf{u} = \mathbf{f} \quad &\text{in } \Omega, \\ \nabla \cdot \mathbf{u} = 0 \quad &\text{in } \Omega, \end{aligned} \tag{1.8}$$

together with appropriate boundary conditions (to be specified later), $\langle \cdot \rangle$ denotes the expected value (mean).

To study the roughness functional effectively, we determine the first-order and second-order shape derivatives, namely, the shape gradient and shape Hessian, using tools from shape calculus. The focus of this work is twofold. First, expressions for the shape gradient and shape Hessian of $\mathscr{D}$ for general boundary roughness are derived. Second, these expressions are evaluated for a perfectly flat boundary and used in a Taylor expansion to study $\mathscr{D}$ for small roughnesses. The dependence of the drag increments on the viscosity ν and on the characteristics of (small) boundary perturbations are then analyzed.

Note that in the lubrication case, the Reynolds number is generally low, and, in many cases, the flow can be modeled as steady. Thus, the Stokes equations or the steady low Reynolds number Navier-Stokes equations provide a good model for studying roughness effects for lubrication flows.

In turbulent flows, the Reynolds number of the flow as a whole is high, but the relevant feature of the flow is the viscous sub-layer, and the Reynolds number of this layer, based on its thickness and the velocity at its edge, is always small, independent of the flow Reynolds number. This occurs because the viscous layer becomes thinner as the Reynolds number increases. Turbulent flows are also unsteady, but the unsteadiness in the viscous sub-layer is driven by turbulence farther away from the wall, which is on time scales that are larger than the viscous time scale that governs sub-layer dynamics. Thus, as a first approximation, a roughness analysis in the steady low Reynolds number Navier-Stokes equations serves as a model for the turbulent case. As a sequel, in later analysis, we first derive the expressions for the shape derivatives in the context of laminar flow, i.e., the velocity field satisfies the stationary Navier-Stokes equation, then discuss extention of this analysis to the turbulent flows.

As stated above, a common approach to analyzing flow over rough boundaries is to homogenize the roughness to find effective boundary conditions (or a rough wall law) to be applied in a smoother domain, e.g., [1, 33, 51]. Parameters in the resulting Robin-type boundary conditions are usually derived numerically from the solution of problems on representative cutouts of the domain. The homogenized problem, which uses the effective boundary conditions, is solved on a smooth domain, and the solution is considered as an approximation to the rough wall flow. In lubrication problems, it is common to homogenize the simpler Reynolds lubrication equation as an alternative to the Navier-Stokes equations [5, 6, 60]. Our approach is significantly different from homogenization since we measure the drag directly through the roughness functional (1.7) and consider the shape as a variable. This allows first- and second-order derivative information to be obtained, which is an attractive feature of the shape calculus-based approach for the analysis of flow over rough surfaces.

In the past, shape Hessians (i.e., second-order shape derivatives) have been used mainly as a tool to analyze the well-posedness of shape optimization problems [9, 10, 20, 28]. More recently, approximations of shape Hessians have also been used to accelerate convergence of iterative methods for solution of shape optimization problems, for example in imaging [29], aerodynamic design [22, 64], and elliptic shape optimization problems [20, 21].

The tools of shape calculus are reviewed briefly in Chapter 2 and are used in Chapter 3 to derive the shape gradient and Hessian of the roughness functional for laminar flows. Results of this analysis are applied to Couette and Poiseuille flow problems in Chapter 4. The extension to turbulent flows is given in Chapter 5, and in Chapter 6, the relevant numerical analysis for the time integration scheme is summarized. Due to the challenge in the sensitivity analysis of turbulent flows, the Lorenz problem is used as a simplified model problem in Chapter 7. Finally conclusion and future work are discussed in Chapter 8.

Chapter 2

Preliminaries

In this chapter basic results from shape calculus are recalled, providing a general framework for the computation of first- and second-order shape derivatives (Section 2.1). Furthermore, when the shape derivatives are to be taken in the presence of equality constraints, in this case imposed by the Navier-Stokes equations, it is convenient to use the Lagrange method to incorporate the constraints. This approach is outlined in Section 2.2.

2.1 Shape gradients and shape Hessians

We use the *velocity method* (see [15, 19, 76]) for the computation of shape derivatives, as summarized below. Let $D \subset \mathbb{R}^N$ be open and $\Omega \subset D$. We consider perturbations $\Omega_t(\mathbf{V}) \subset D$ of Ω defined through a perturbation of identity, i.e,

$$\Omega_t(\mathbf{V}) = \{\mathbf{x} + t\mathbf{V}(\mathbf{x}); \mathbf{x} \in \Omega\}, \quad |t| \leq t_o,$$

where $\mathbf{V}$ is a Lipschitz continuous velocity field[1]. Let $\mathcal{J}(\cdot) : \Omega_t(\mathbf{V}) \to \mathbb{R}$ be a functional, then $\mathcal{J}$ is said to have a *Hadamard semiderivative* at Ω in direction $\mathbf{V}$, if the

---

[1] Note that the velocity field discussed here is a velocity associated with the continuous transformation of the solution domain in pseudo-time t. It should not be confused with the fluid velocity $\mathbf{u}$.

limit

$$d\mathcal{J}(\Omega;\mathbf{V}) := \lim_{t\to 0} \frac{\mathcal{J}(\Omega_t(\mathbf{V})) - \mathcal{J}(\Omega)}{t} = \frac{d}{dt}\mathcal{J}(\Omega_t)\bigg|_{t=0}$$

exists and is finite.

For later use, we now give derivatives for two shape functionals.

Lemma 2.1.1. *Let $\Omega \subset D$ be a bounded and measurable domain and $\mathbf{V}(\mathbf{x})$ a Lipschitz continuous velocity field. Then, for $\psi \in W^{1,1}(D) \cap L^1(D)$, $\phi \in H^2(D)$, the derivatives of the shape functionals*

$$\mathcal{I}(\Omega_t(\mathbf{V})) = \int_{\Omega_t(\mathbf{V})} \psi \, d\mathbf{x}, \qquad \mathcal{J}(\Omega_t(\mathbf{V})) = \int_{\partial\Omega_t(\mathbf{V})} \phi \, d\mathbf{s},$$

are given by:

$$d\mathcal{I}(\Omega;\mathbf{V}) = \frac{d}{dt}\mathcal{I}(\Omega_t)\bigg|_{t=0} = \int_{\partial\Omega} \psi \mathbf{V}\cdot\mathbf{n}\, d\mathbf{s}, \tag{2.1}$$

$$d\mathcal{J}(\Omega;\mathbf{V}) = \frac{d}{dt}\mathcal{J}(\Omega_t)\bigg|_{t=0} = \int_{\partial\Omega} (\nabla\phi\cdot\mathbf{n} + H\phi)\mathbf{V}\cdot\mathbf{n}\, d\mathbf{s}, \tag{2.2}$$

where H is the mean curvature of the boundary and $\mathbf{n}$ is the unit outward normal vector of $\partial\Omega$.

A detailed proof in a more general setting, which is based on a change of variables and the chain rule, can be found, e.g., in [19, pages 352–355]. Now, for $\mathbf{g} \in W^{2,1}(D) \cap H^1(D)$, consider the shape functional

$$\mathcal{K}(\Omega_t(\mathbf{V})) = \int_{\partial\Omega_t(\mathbf{V})} \mathbf{g}\cdot\mathbf{n}\, d\mathbf{s}.$$

Using Lemma 2.1.1, the shape derivative of $\mathcal{K}$ is

$$\begin{aligned} d\mathcal{K}(\Omega;\mathbf{V}) = \frac{d}{dt}\mathcal{K}(\Omega_t)\Big|_{t=0} &= \frac{d}{dt}\left(\int_{\partial\Omega_t(\mathbf{V})} \mathbf{g}\cdot\mathbf{n}\,d\mathbf{s}\right)\Big|_{t=0} \\ &= \frac{d}{dt}\left(\int_{\Omega_t(\mathbf{V})} \nabla\cdot\mathbf{g}\,d\mathbf{x}\right)\Big|_{t=0} \overset{\text{(using 2.1)}}{=} \int_{\partial\Omega} \nabla\cdot\mathbf{g}\,\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}, \end{aligned} \tag{2.3}$$

in which the boundary integral is transformed into an integral over the domain Ω so that (2.1) can be applied. This technique is also useful for the derivation of second-order shape derivatives, which are based on a second transformation of the domain $\Omega_t(\mathbf{V})$, namely on

$$\Omega_{t,r}(\mathbf{V},\mathbf{W}) = \{\mathbf{x}+t\mathbf{V}(\mathbf{x})+r\mathbf{W}(\mathbf{x}+t\mathbf{V}(\mathbf{x}));\, \mathbf{x}\in\Omega\}, \quad |t|\le t_0, |r|\le r_0,$$

where $\mathbf{V}$ and $\mathbf{W}$ denote sufficiently smooth velocity fields. The second-order shape derivative of a functional $\mathcal{J}$ is then defined by

$$d^2\mathcal{J}(\Omega;\mathbf{V},\mathbf{W}) = \frac{\partial}{\partial r}\left(\frac{\partial}{\partial t}\mathcal{J}_{\mathbf{V},\mathbf{W}}(t,r)\Big|_{t=0}\right)\Big|_{r=0}.$$

For $f\in H^2(D)$, consider the functional

$$\mathcal{J}_{\mathbf{V},\mathbf{W}}(t,r) = \mathcal{J}(\Omega_{t,r}(\mathbf{V},\mathbf{W})) := \int_{\Omega_{t,r}(\mathbf{V},\mathbf{W})} f\,d\mathbf{x}.$$

Using the expression for shape gradients,

$$\frac{\partial}{\partial t}\mathcal{J}_{\mathbf{V},\mathbf{W}}(t,r)\Big|_{t=0} = d\mathcal{J}(\Omega_r(\mathbf{W});\mathbf{V}) = \int_{\partial\Omega_r(\mathbf{W})} f\,\mathbf{V}\cdot\mathbf{n}_r\,d\mathbf{s},$$

where $\mathbf{n}_r$ is here the outward normal vector of $\partial\Omega_r$. Using (2.3), the second shape derivative is given by:

$$\begin{aligned} d^2\mathcal{J}(\Omega;\mathbf{V},\mathbf{W}) &= \frac{d}{dr}\left(\int_{\partial\Omega_r(\mathbf{W})} f\mathbf{V}\cdot\mathbf{n}_r\,d\mathbf{s}\right) = \frac{d}{dr}\left(\int_{\Omega_r(\mathbf{W})} \nabla\cdot(f\mathbf{V})\,d\mathbf{x}\right) \\ &= \int_{\partial\Omega} \nabla\cdot(f\mathbf{V})(\mathbf{W}\cdot\mathbf{n})\,d\mathbf{s}. \end{aligned} \tag{2.4}$$

Naturally, the question arises under which conditions the shape Hessian is symmetric. As shown in [19], a sufficient condition for the shape Hessian to be symmetric is that the velocity fields **V** and **W** satisfy

$$\int_{\partial\Omega} f\left((\nabla\mathbf{V})\mathbf{W} - (\nabla\mathbf{W})\mathbf{V}\right) \cdot \mathbf{n}\, d\mathbf{s} = 0 \quad \text{for all } f \in C^2(D). \tag{2.5}$$

Note that in the shape derivative formulas stated here (2.1, 2.2, 2.3, 2.4), the derivatives are expressed in terms of boundary integrals involving **V** and **W**, so that the shape perturbations only matter in a neighborhood of the boundary $\partial\Omega$.

2.2 The formal Lagrange method

We seek the shape derivatives in the presence of an equality constraint (here the Navier-Stokes equations). The constraint can be incorporated using Lagrange multipliers. This approach is outlined below, but without concern for the technical details that arise in infinite dimensions (see [18, 25, 32, 47] for a more complete treatment). We are interested in the derivative of a functional $J(q,u)$ with respect to a *control variable* q, in which the *state variable* u depends implicitly on q through a *state equation* $c(q,u) = 0$:

$$\begin{gathered} \hat{J}: \quad q \longmapsto J(q,u) \\ \text{s. t.} \\ c(q,u) = 0. \end{gathered} \tag{2.6}$$

Here, $\hat{J}$ incorporates the dependence of the state u on the control q via the solution of $c(q,u) = 0$; in other words, $\hat{J}(q) := J(q,u(q))$.

Introducing the Lagrange multiplier (also known as the *adjoint state variable*) λ, the *Lagrangian functional* is defined by

$$\mathscr{L}(q,u,\lambda) = J(q,u) + (c(q,u),\lambda),$$

where $(\cdot,\cdot)$ denotes an appropriate inner product. The gradient of the Lagrangian functional is given by:

$$\begin{bmatrix} \delta_u \mathscr{L} \\ \delta_q \mathscr{L} \\ \delta_\lambda \mathscr{L} \end{bmatrix} := \begin{bmatrix} \delta_u J + c_u^\star \lambda \\ \delta_q J + c_q^\star \lambda \\ c \end{bmatrix},$$

where $\delta_u \mathscr{L}, \delta_q \mathscr{L}$ are the variations of the Lagrangian with respect to u and q, respectively, and analogously for $\delta_u J$ and $\delta_q J$, and the dependence of all operators on q and u has been suppressed for notational simplicity. Moreover, c_u and c_q are the Jacobians of the state equations with respect to the state and control variables, respectively, and $c_u^\star$ and $c_q^\star$ denote the adjoint operators. The *reduced gradient* $d\hat{J}/dq$ is then given simply by $\delta_q \mathscr{L}$ evaluated for values of u and λ such that the *adjoint equation* $\delta_u \mathscr{L} = 0$ and the state equation $\delta_\lambda \mathscr{L} = 0$ are satisfied.

The reduced gradient can thus be determined by the following procedure:

1. Given q, solve the state equation $c = 0$ for u.
2. Given q and u, solve the adjoint equation $\delta_u J + c_u^* \lambda = 0$ for λ.
3. Given q, u, and λ, evaluate the reduced gradient: $d\hat{J}/dq = \delta_q J + c_q^* \lambda$.

The *reduced Hessian* $d^2\hat{J}/dq^2$ can be determined by finding the Schur complement of the control block in the matrix of second variations of $\mathscr{L}$ with respect to (u, q, λ), i.e., in the matrix

$$\begin{bmatrix} \delta^2_{uu}\mathscr{L} & \delta^2_{uq}\mathscr{L} & c^*_u \\ \delta^2_{qu}\mathscr{L} & \delta^2_{qq}\mathscr{L} & c^*_q \\ c_u & c_q & 0 \end{bmatrix}.$$

Thus,

$$\frac{d^2\hat{J}}{dq^2} = \delta^2_{qq}\mathscr{L} - \begin{bmatrix} \delta^2_{qu}\mathscr{L} & c^*_q \end{bmatrix} \begin{bmatrix} \delta^2_{uu}\mathscr{L} & c^*_u \\ c_u & 0 \end{bmatrix}^{-1} \begin{bmatrix} \delta^2_{uq}\mathscr{L} \\ c_q \end{bmatrix}, \tag{2.7}$$

where $\delta^2_{uu}\mathscr{L}$ and $\delta^2_{uq}\mathscr{L}$ are the variations of $\delta_u\mathscr{L}$ with respect to u and q, respectively, and analogously for $\delta^2_{qu}\mathscr{L}$, $\delta^2_{qq}\mathscr{L}$. Often, it is not necessary or feasible to fully construct the reduced Hessian. Instead, it suffices to know its action in arbitrary directions. An efficient way to compute the action of the reduced Hessian in a direction $\hat{q}$ is to introduce auxiliary variables $\hat{u}, \hat{\lambda}$ defined by

$$-\begin{bmatrix} \delta^2_{uu}\mathscr{L} & c^*_u \\ c_u & 0 \end{bmatrix}^{-1} \begin{bmatrix} \delta^2_{uq}\mathscr{L} \\ c_q \end{bmatrix} \hat{q} = \begin{bmatrix} \hat{u} \\ \hat{\lambda} \end{bmatrix}. \tag{2.8}$$

In the sequel, $\hat{u}$ and $\hat{\lambda}$ are referred to as *incremental state* and *incremental adjoint* variables. Once the state and adjoint equations have been solved, the action of the reduced Hessian in the direction $\hat{q}$ can be computed as follows:

1. Given q, u and $\hat{q}$, solve the *incremental state equation* (i.e., the lower equation in (2.8))

   $$c_u\hat{u} + c_q\hat{q} = 0 \quad \text{for } \hat{u}.$$

2. Given $q, u, \lambda, \hat{q}$ and $\hat{u}$, solve the *incremental adjoint equation* (i.e., the upper equation in (2.8))

   $$\delta^2_{uu}\mathscr{L}\hat{u} + c^*_u\hat{\lambda} + \delta^2_{uq}\mathscr{L}\hat{q} = 0 \quad \text{for } \hat{\lambda}.$$

3. Finally, given $q, u, \lambda, \hat{q}, \hat{u}, \hat{\lambda}$ evaluate the action of the reduced Hessian on $\hat{q}$ by

$$\frac{d^2\hat{J}}{dq^2}\hat{q} = \delta^2_{qq}\mathscr{L}\hat{q} + \delta^2_{qu}\mathscr{L}\hat{u} + c^*_q\hat{\lambda}.$$

Note that the incremental state equation can be viewed as one more variation of $\delta_\lambda\mathscr{L}$ with respect to all variables; thus, below we will use $\delta^2_{\lambda\bullet}$ to denote the residual of the incremental state equation, and analogously $\delta^2_{u\bullet}$ for the incremental adjoint equation.

Chapter 3

The roughness functional and its shape derivatives for stationary flows

The shape calculus outlined in Chapter 2 will be applied here to the roughness functional that maps the boundary topography to the drag exerted on fluid flow governed by the stationary Navier-Stokes equations. The resulting expressions for the shape gradient and shape Hessian will be evaluated in Chapter 4 to analyze the effects of small roughness on an otherwise flat boundary. The extension to the turbulent cases will be discussed later.

3.1 Problem statement

Of interest here is the steady incompressible flow of a fluid between two walls (e.g. in a channel), where for simplicity only one wall is rough, as sketched in Figure 3.1. The flow is governed by the incompressible, stationary Navier-Stokes equations

$$\begin{aligned} -\nabla\cdot\big(-\mathbf{I}p+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\big)+\mathbf{u}\cdot\nabla\mathbf{u}-\mathbf{f}=\mathbf{0} \qquad &\text{in } \Omega, \\ \nabla\cdot\mathbf{u}=0 \qquad &\text{in } \Omega, \end{aligned} \tag{3.1a}$$

on the domain $\Omega\subset D\subset\mathbb{R}^N$, $(N=2,3)$. Here, $\mathbf{u}$ is the flow velocity, which should not be confused with the domain deformation "velocities" $\mathbf{V}$ and $\mathbf{W}$, p is the pres-

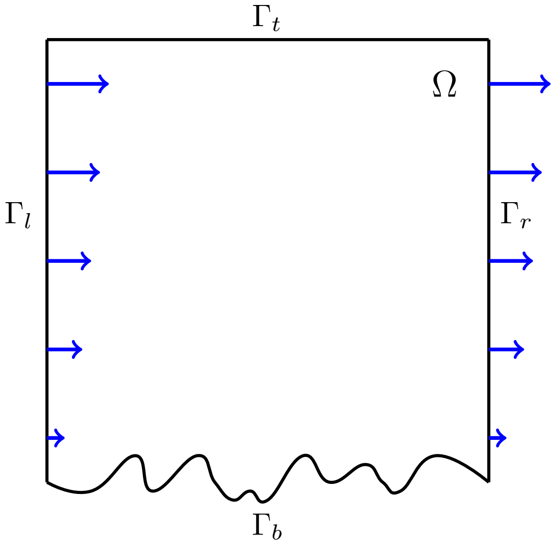

Figure 3.1: Sketch of domain Ω with rough bottom boundary Γ_b. The flow is periodic on the left and right boundaries Γ_l and Γ_r. In three dimensions, the flow is also periodic on the fore and aft boundaries Γ_f and Γ_a not shown.

sure and $\mathbf{f}$ is a body force. The latter two variables have been rescaled to eliminate the density. The channel is taken to be infinitely long in the directions parallel to the wall, with roughness topography, boundary condition data and forcing $\mathbf{f}$ that are periodic in the streamwise (x) and spanwise (y) directions with periods L_x and L_y respectively. The solutions we seek are also periodic in both directions with the same periods. Solutions will thus be sought on a domain Ω that is finite with periodic boundary conditions in the streamwise and spanwise directions. To be precise, in case of a three-dimensional domain Ω (i.e. $N = 3$; specialization to $N = 2$ is obvious) the boundaries in the x direction (left and right) and the y direction (front and aft) are denoted by $\Gamma_l, \Gamma_r, \Gamma_f$ and Γ_a, respectively, and the flow velocity $\mathbf{u}$ as well as the traction $\left(-\mathrm{I}\,p + \nu(\nabla\mathbf{u} + \nabla\mathbf{u}^T)\right)\mathbf{n}$ (where $\mathbf{n}$ is the unit outward normal vector) must coincide on Γ_l and Γ_r, and on Γ_f and Γ_a. This, along with $\nabla \cdot \mathbf{u} = 0$, implies that p and $\nabla \mathbf{u}$ must also coincide on periodic boundaries. For subsequent use, the periodic boundaries are designated collectively as Γ_p and vector fields such as $\mathbf{u}$ will simply be described as periodic on Γ_p when these conditions are satisfied. For the bottom and top boundaries Γ_b and Γ_t, respectively, the no-slip boundary condition is imposed, so the velocity is specified as

$$\mathbf{u} = \mathbf{0} \qquad \text{on } \Gamma_b, \tag{3.1b}$$

$$\mathbf{u} = \mathbf{u}_0 \qquad \text{on } \Gamma_t, \tag{3.1c}$$

where $\mathbf{u}_0$ is the specified velocity of the upper boundary with $\mathbf{u}_0 \cdot \mathbf{n} = 0$ on Γ_t.

Our primary interest is the effect of roughness on the drag for a domain with fixed volume. In this context, the drag is simply the driving force required to maintain the steady fluid flow, and it is generally of interest because energy is required to do work on the flow. The rate at which work is done on the flow is thus a particularly relevant measure of the drag phenomenon. Furthermore, in the special cases of Couette flow ($\mathbf{f} = 0$, $\mathbf{u}_0$ fixed) and Poiseuille flow ($\mathbf{u}_0 = 0$, volume flow rate fixed) to be considered in Chapter 4, the drag force is directly proportional to the rate of work. Using the notation $\boldsymbol{\tau} = -\mathrm{I}p + \nu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)$ for the stress tensor, the total rate of work on the fluid is given by:

$$\begin{aligned}
&\int_{\partial\Omega} \mathbf{u} \cdot (\boldsymbol{\tau}\mathbf{n})\, d\mathbf{s} + \int_{\Omega} \mathbf{u} \cdot \mathbf{f}\, d\mathbf{x} \\
&= \int_{\Omega} \nabla \mathbf{u} : \boldsymbol{\tau}\, d\mathbf{x} + \int_{\Omega} \mathbf{u} \cdot (\nabla \cdot \boldsymbol{\tau} + \mathbf{f})\, d\mathbf{x} \\
&= \int_{\Omega} \frac{\nu}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^T) : (\nabla \mathbf{u} + \nabla \mathbf{u}^T)\, d\mathbf{x} + \int_{\Omega} \mathbf{u} \cdot (\mathbf{u} \cdot \nabla \mathbf{u})\, d\mathbf{x} \\
&= \int_{\Omega} \frac{\nu}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^T) : (\nabla \mathbf{u} + \nabla \mathbf{u}^T)\, d\mathbf{x}.
\end{aligned}$$

The last equality follows from

$$\int_\Omega \mathbf{u}\cdot(\mathbf{u}\cdot\nabla\mathbf{u})\,d\mathbf{x} = \frac{1}{2}\int_{\partial\Omega}(\mathbf{u}\cdot\mathbf{u})\mathbf{u}\cdot\mathbf{n}\,d\mathbf{s} - \frac{1}{2}\int_\Omega(\mathbf{u}\cdot\mathbf{u})\nabla\cdot\mathbf{u}\,d\mathbf{x} = 0,$$

which holds since $\mathbf{u}\cdot\mathbf{n} = 0$ on Γ_b and Γ_t, periodicity holds on Γ_p and $\nabla\cdot\mathbf{u} = 0$. Thus, the effect of roughness on the drag is the same as its effect on the rate of kinetic energy dissipation. The roughness functional $\mathscr{D}$ will be defined using the dissipation form because this is more convenient for the subsequent derivations. Note that, to obtain a certain structure of the adjoint Navier-Stokes equations (which will be derived below), the work done by drag is scaled by a factor of $1/2$ in $\mathscr{D}$:

$$\begin{aligned}&\mathscr{D}(\Gamma_b) = \frac{\nu}{4}\int_\Omega(\nabla\mathbf{u}+\nabla\mathbf{u}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\,d\mathbf{x},\\ &\text{where } (\mathbf{u},p) \text{ solves (3.1) on } \Omega.\end{aligned} \tag{3.2}$$

The shape derivatives of the roughness functional $\mathscr{D}$ can now be derived. Since the roughness functional (3.2) involves the solution of the Navier-Stokes equations, the Lagrangian approach described in Section 2.2 is used. For that purpose, the Navier-Stokes equations are written in variational form and treated as equality constraints in appropriate infinite-dimensional spaces. In particular, consider the following subspaces of $\mathbf{H}^1(\Omega) := (H^1(\Omega))^N$:

$$\begin{aligned}\tilde{\mathbf{H}}^1(\Omega) &= \{\mathbf{u}\in\mathbf{H}^1(\Omega) : \mathbf{u} \text{ is periodic on } \Gamma_p\},\\ \tilde{\mathbf{H}}^1_{\mathbf{u}_0}(\Omega) &= \{\mathbf{u}\in\tilde{\mathbf{H}}^1(\Omega) : \mathbf{u} = \mathbf{u}_0 \text{ on } \Gamma_t\},\\ \tilde{\mathbf{H}}^1_0(\Omega) &= \{\mathbf{u}\in\tilde{\mathbf{H}}^1(\Omega) : \mathbf{u} = \mathbf{0} \text{ on } \Gamma_t\},\end{aligned}$$

where $\mathbf{u}_0 \in \tilde{\mathbf{H}}^1(\Omega)$ with $\mathbf{u}_0\cdot\mathbf{n} = 0$ on Γ_t. The variational form of the Navier-Stokes equation is given as follows: Find $(\mathbf{u},p)\in\tilde{\mathbf{H}}^1_{\mathbf{u}_0}(\Omega)\times L^2(\Omega)$ such that

$$\int_\Omega \frac{\nu}{2}(\nabla\mathbf{u}+\nabla\mathbf{u}^T):(\nabla\mathbf{v}+\nabla\mathbf{v}^T)\,d\mathbf{x} - \int_\Omega(q\nabla\cdot\mathbf{u}+p\nabla\cdot\mathbf{v})\,d\mathbf{x} + \int_\Omega(\mathbf{u}\cdot\nabla\mathbf{u})\cdot\mathbf{v}\,d\mathbf{x}$$

$$-\int_{\Omega}\mathbf{f}\cdot\mathbf{v}\,d\mathbf{x}+\int_{\Gamma_b}\boldsymbol{\alpha}\cdot\mathbf{u}\,d\mathbf{s}=\int_{\Gamma_b}\mathbf{v}\cdot\left(-\mathrm{I}p+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\right)\mathbf{n}\,d\mathbf{s} \tag{3.3}$$

for all $(\mathbf{v},q,\boldsymbol{\alpha})\in\tilde{\mathbf{H}}_0^1(\Omega)\times L^2(\Omega)\times\mathbf{H}^{-1/2}(\Gamma_b)$. Note that in (3.3) the boundary condition on Γ_b is enforced weakly as this is needed for the computation of shape derivatives. It is well known (see, e.g., [69], page 67) that (3.3) admits a solution $(\mathbf{u},p)\in\tilde{\mathbf{H}}^1_{\mathbf{u}_0}(\Omega)\times L^2(\Omega)$ for each Ω with $\partial\Omega\in C^{0,1}$ and $\mathbf{f}\in\mathbf{L}^2(\Omega)$. Here, higher regularity $\mathbf{f}\in\tilde{\mathbf{H}}^2(\Omega)$ (i.e., $\mathbf{f}\in\mathbf{H}^2(\Omega)$ and $\mathbf{f}$ is periodic on Γ_p) and bounded Ω with $\partial\Omega\in C^{3,1}$ are assumed, to meet requirements for the existence of shape derivatives. Introducing Lagrange multipliers $\boldsymbol{\lambda}\in\tilde{\mathbf{H}}_0^1(\Omega)$, $\alpha\in L^2(\Omega)$ and $\boldsymbol{\beta}\in\mathbf{H}^{-1/2}(\Gamma_b)$, the Lagrangian functional is defined by

$$\begin{aligned}\mathscr{L}(\Gamma_b,\mathbf{u},p,\boldsymbol{\lambda},\alpha,\boldsymbol{\beta})=&\frac{\nu}{4}\int_{\Omega}(\nabla\mathbf{u}+\nabla\mathbf{u}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\,d\mathbf{x}-\int_{\Omega}\alpha\nabla\cdot\mathbf{u}+\nabla\cdot\boldsymbol{\lambda}p\,d\mathbf{x}\\&+\int_{\Omega}\left(\frac{\nu}{2}(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)+(\mathbf{u}\cdot\nabla\mathbf{u})\cdot\boldsymbol{\lambda}-\boldsymbol{\lambda}\cdot\mathbf{f}\right)d\mathbf{x}\\&-\int_{\Gamma_b}\boldsymbol{\lambda}\cdot\left(-\mathrm{I}p+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\right)\mathbf{n}\,d\mathbf{s}+\int_{\Gamma_b}\boldsymbol{\beta}\cdot\mathbf{u}\,d\mathbf{s}.\end{aligned}$$

In the derivations of the shape gradient and shape Hessian, $\mathbf{u},p$ are referred to as state variables, $\boldsymbol{\lambda},\alpha,\boldsymbol{\beta}$ are referred to as adjoint variables, and the incremental state and incremental adjoint variables are denoted by $\hat{\mathbf{u}},\hat{p},\hat{\boldsymbol{\lambda}},\hat{\alpha},\hat{\boldsymbol{\beta}}$. Finally, test functions in variational formulations are denoted by $\tilde{\mathbf{u}},\tilde{p},\tilde{\boldsymbol{\lambda}},\tilde{\alpha},\tilde{\boldsymbol{\beta}}$.

3.2 Shape gradient

As described in Section 2.2, determining the gradient of $\mathscr{D}$ with respect to the shape Ω (or equivalently Γ_b) requires solutions to the state and adjoint equations. It is easy to see that, setting variations of $\mathscr{L}$ with respect to the Lagrange multipliers to zero results in the weak form (3.3) of the Navier-Stokes equations. Variations with respect to **u** and p result in the adjoint equations for $(\boldsymbol{\lambda}, \alpha)$:

$$\begin{aligned}\delta_{\mathbf{u}}\mathscr{L} = & \int_{\Omega} \Big(\frac{\nu}{2}(\nabla\mathbf{u}+\nabla\mathbf{u}^T) : (\nabla\tilde{\mathbf{u}}+\nabla\tilde{\mathbf{u}}^T) + \frac{\nu}{2}(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T) : (\nabla\tilde{\mathbf{u}}+\nabla\tilde{\mathbf{u}}^T)\Big)\, d\mathbf{x} \\ & - \int_{\Omega} \alpha\nabla\cdot\tilde{\mathbf{u}}\, d\mathbf{x} - \int_{\Gamma_b} \boldsymbol{\lambda}\cdot\nu(\nabla\tilde{\mathbf{u}}+\nabla\tilde{\mathbf{u}}^T)\mathbf{n}\, d\mathbf{s} + \int_{\Gamma_b} \boldsymbol{\beta}\cdot\tilde{\mathbf{u}}\, d\mathbf{s} \qquad (3.4a) \\ & + \int_{\Omega} (\tilde{\mathbf{u}}\cdot\nabla\mathbf{u}+\mathbf{u}\cdot\nabla\tilde{\mathbf{u}})\cdot\boldsymbol{\lambda}\, d\mathbf{x} = 0, \\ \delta_p\mathscr{L} = & \int_{\Omega} (-\nabla\cdot\boldsymbol{\lambda})\tilde{p}\, d\mathbf{x} + \int_{\Gamma_b} \tilde{p}\boldsymbol{\lambda}\cdot\mathbf{n}\, d\mathbf{s} = 0 \qquad (3.4b)\end{aligned}$$

for all $(\tilde{\mathbf{u}}, \tilde{p}) \in \tilde{\mathbf{H}}_0^1(\Omega) \times L^2(\Omega)$. Integration by parts in (3.4a) yields

$$\begin{aligned}& \int_{\Omega} \Big(-\nabla\cdot\Big(-I\alpha+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)+\nu(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T)\Big)+\nabla\mathbf{u}^T\boldsymbol{\lambda}-\nabla\boldsymbol{\lambda}\mathbf{u}\Big)\cdot\tilde{\mathbf{u}}\, d\mathbf{x} \\ & + \int_{\partial\Omega} (-I\alpha+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)+\nu(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T))\mathbf{n}\cdot\tilde{\mathbf{u}}\, d\mathbf{s} + \int_{\Gamma_b} \boldsymbol{\beta}\cdot\tilde{\mathbf{u}}\, d\mathbf{s} \\ & - \int_{\partial\Omega} (\mathbf{u}\cdot\mathbf{n})(\boldsymbol{\lambda}\cdot\tilde{\mathbf{u}})\, d\mathbf{s} - \int_{\partial\Omega} \boldsymbol{\lambda}\cdot(\nabla\tilde{\mathbf{u}}+\nabla\tilde{\mathbf{u}}^T)\mathbf{n}\, d\mathbf{s} = 0. \qquad (3.5)\end{aligned}$$

From the boundary condition for **u**, we conclude that the corresponding strong form of the adjoint equations is given by:

$$-\nabla\cdot\Big(-\mathrm{I}\,\alpha+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)+\nu(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T)\Big)+\nabla\mathbf{u}^T\boldsymbol{\lambda}-\nabla\boldsymbol{\lambda}\mathbf{u}=\mathbf{0} \text{ in } \Omega, \quad (3.6a)$$

$$\nabla\cdot\boldsymbol{\lambda} = 0 \text{ in } \Omega, \quad (3.6b)$$

with $\boldsymbol{\lambda} = 0$ on $\Gamma_t \cup \Gamma_b$ and periodic on Γ_p. In addition (3.5) implies that

$$\boldsymbol{\beta} = -\Big(-\mathrm{I}\,\alpha+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)+\nu(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T)\Big)\mathbf{n}. \qquad (3.6c)$$

Due to the regularity assumptions on **f** and $\partial\Omega$, regularity theory (see, e.g., [68]) yields that $\mathbf{u} \in \mathbf{H}^4(\Omega) \cap \mathbf{H}_0^1(\Omega)$ and $p \in H^3(\Omega)$. Due to our assumption $\Omega \in$

$C^{3,1}$, $\mathbf{u}$ and p can be extended to functions in $\mathbf{H}^4(D)$ and $H^3(D)$, respectively (see [23]). Similar results hold for $\boldsymbol{\lambda}$ and α, and (3.6c) implies that $\boldsymbol{\beta} \in \mathbf{H}^3(D)$. Thus, the integrands in $\mathscr{L}$ satisfy the requirements of Lemma 2.1.1, which shows that the Lagrangian $\mathscr{L}$ is shape differentiable with respect to Γ_b. Additionally, for $N = 2,3$, the Rellich-Kondrachov embedding theorem (see [2, page 144]), shows that $\mathbf{f} \in C(\bar{\Omega})$ and $\mathbf{u} \in C^2(\bar{\Omega})$, where $\bar{\Omega}$ is the closure of Ω. This regularity result implies that the Navier-Stokes equations are satisfied up to the boundary, which allows certain simplifications of boundary terms.

Next, the velocity method is used to derive the shape derivative in direction $\mathbf{V}$ via (2.1), (2.2) and (2.3). To apply (2.3) for the computation of the derivative with respect to the shape, the boundary integrals over Γ_b in the Lagrangian functional can be extended to $\partial\Omega$, if $\boldsymbol{\beta}$ is extended to Ω such that it is periodic on Γ_p and vanishes on Γ_t. Thus, the shape derivative is given by:

$$
\begin{aligned}
&d\mathscr{D}(\Gamma_b;\mathbf{V})\\
&\quad=\int_{\Gamma_b} \frac{\nu}{4}(\nabla\mathbf{u}+\nabla\mathbf{u}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s} - \int_{\Gamma_b} \alpha\nabla\cdot\mathbf{u}\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}\\
&\qquad+\int_{\Gamma_b} \Big(-\nabla\cdot\boldsymbol{\lambda}p+\frac{\nu}{2}(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)-\boldsymbol{\lambda}\cdot\mathbf{f}\Big)\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}\\
&\qquad-\int_{\Gamma_b} \nabla\cdot\big(\boldsymbol{\lambda}\cdot\big(-\mathrm{I}p+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\big)\big)\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}+\int_{\Gamma_b}(\mathbf{u}\cdot\nabla\mathbf{u})\cdot\boldsymbol{\lambda}\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}\\
&\qquad+\int_{\Gamma_b} \nabla\cdot\Big(-\Big(-\mathrm{I}\alpha+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)+\nu(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T)\Big)\mathbf{u}\Big)\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}\\
&\quad=\int_{\Gamma_b}\Big(\frac{\nu}{4}(\nabla\mathbf{u}+\nabla\mathbf{u}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)+\frac{\nu}{2}(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\Big)\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}\\
&\qquad-\int_{\Gamma_b}\Big(\nabla\mathbf{u}:\Big(\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)+\nu(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T)\Big)+\nabla\boldsymbol{\lambda}:\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\Big)\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}\\
&\qquad+\int_{\Gamma_b}\boldsymbol{\lambda}\cdot\big(-\nabla\cdot\big(-\mathrm{I}p+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\big)+\mathbf{u}\cdot\nabla\mathbf{u}-\mathbf{f}\big)\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}
\end{aligned}
$$

$$+\int_{\Gamma_b} -\nabla\cdot\Big(-\mathrm{I}\alpha+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)+\nu(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T)\Big)\cdot\mathbf{u}\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}$$

$$=\int_{\Gamma_b}\Big(-\frac{\nu}{4}(\nabla\mathbf{u}+\nabla\mathbf{u}^T)-\frac{\nu}{2}(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T)\Big):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}, \tag{3.7}$$

where $\mathbf{V}$ is the boundary deformation for which the gradient is evaluated, and the regularity properties argued above, as well as the equality

$$\nabla\mathbf{u}:\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)=\frac{\nu}{2}(\nabla\mathbf{u}+\nabla\mathbf{u}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T),$$

are used to simplify the expressions. Note that, compared to the approach used in [64], our results for the shape gradient contain additional terms that involve $\nabla\mathbf{u}^T$ and $\nabla\boldsymbol{\lambda}^T$. However, the two expressions can be shown to be equivalent: since $\mathbf{u}$ vanishes on Γ_b, $\nabla\mathbf{u}$ is nonzero only in directions normal to the boundary. The divergence-free condition additionally implies that $(\mathbf{n}\cdot\nabla\mathbf{u})\cdot\mathbf{n}=0$, which yields that

$$\nabla\mathbf{u}:\nabla\mathbf{u}^T=0 \text{ on } \Gamma_b. \tag{3.8}$$

Analogously, one may argue that $\nabla\mathbf{u}:\nabla\boldsymbol{\lambda}^T=0$, and thus our results for the gradient are consistent with those in [64]. The findings in this section are summarized in the next theorem.

Theorem 3.2.1. *The shape gradient of $\mathscr{D}$ at Ω in the direction of the shape velocity field $\mathbf{V}$ is given by* (3.7)*, where $(\mathbf{u},p)$ denote the solution to the state equation* (3.1)*, and $(\boldsymbol{\lambda},\alpha)$ the solution to the adjoint equations* (3.6).

3.3 Shape Hessian

The shape Hessian operator is particularly important at stationary points, where the shape gradient vanishes and the second order derivatives determine the local behavior of the roughness functional. Using the assumptions that $\mathbf{f} \in \mathbf{H}^2(\Omega)$ and Ω is a bounded domain with a $C^{3,1}$ boundary, $d\mathscr{D}(\Gamma_b;\mathbf{V})$ as defined in (3.7) is differentiable with respect to Γ_b since the integrands are in $W^{2,1}(\Omega)$ and can be extended to $W^{2,1}(D)$. Hence one more derivative of the Lagrangian functional can be taken to derive the shape Hessian operator. Let $\mathbf{W}$ be the velocity field corresponding to the second shape variation. Variations of $\delta_{\boldsymbol{\lambda}}\mathscr{L}$ yield that for all $\tilde{\boldsymbol{\lambda}} \in \tilde{\mathbf{H}}_0^1(\Omega)$

$$\begin{aligned}\delta^2_{\boldsymbol{\lambda}\bullet}\mathscr{D}(\Gamma_b,\mathbf{W}) = &\int_\Omega \Big(-\nabla\cdot\tilde{\boldsymbol{\lambda}}\hat{p} + \frac{\nu}{2}(\nabla\tilde{\boldsymbol{\lambda}} + \nabla\tilde{\boldsymbol{\lambda}}^T) : (\nabla\hat{\mathbf{u}} + \nabla\hat{\mathbf{u}}^T) \\ &+ (\hat{\mathbf{u}}\cdot\nabla\mathbf{u} + \mathbf{u}\cdot\nabla\hat{\mathbf{u}})\cdot\tilde{\boldsymbol{\lambda}}\Big)\,d\mathbf{x} - \int_{\Gamma_b} \tilde{\boldsymbol{\lambda}}\cdot\big(-\mathrm{I}\hat{p} + \nu(\nabla\hat{\mathbf{u}} + \nabla\hat{\mathbf{u}}^T)\big)\,\mathbf{n}\,d\mathbf{s} \\ &+ \int_{\Gamma_b} \big(-\nabla\cdot\big(-\mathrm{I}p + \nu(\nabla\mathbf{u} + \nabla\mathbf{u}^T)\big) + \mathbf{u}\cdot\nabla\mathbf{u} - \mathbf{f}\big)\cdot\tilde{\boldsymbol{\lambda}}\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s}.\end{aligned} \tag{3.9}$$

As introduced in Section 2.2, here $\delta^2_{\boldsymbol{\lambda}\bullet}\mathscr{L}$ denotes a variation of $\delta_{\boldsymbol{\lambda}}\mathscr{L}$ with respect to all variables. Using (3.6c), for all $(\tilde{\alpha},\tilde{\boldsymbol{\beta}}) \in L^2(\Omega)\times\mathbf{H}^{-1/2}(\Omega)$,

$$\delta^2_{\alpha\bullet}\mathscr{D}(\Gamma_b,\mathbf{W}) = -\int_\Omega \tilde{\alpha}\nabla\cdot\hat{\mathbf{u}}\,d\mathbf{x} - \int_{\Gamma_b} (\tilde{\alpha}\nabla\cdot\mathbf{u})\mathbf{W}\cdot n\,d\mathbf{s}, \tag{3.10}$$

$$\begin{aligned}\delta^2_{\boldsymbol{\beta}\bullet}\mathscr{D}(\Gamma_b,\mathbf{W}) = &\int_{\Gamma_b} \Big(-\mathrm{I}\tilde{\alpha} + \nu(\nabla\tilde{\mathbf{u}} + \nabla\tilde{\mathbf{u}}^T) + \nu(\nabla\tilde{\boldsymbol{\lambda}} + \nabla\tilde{\boldsymbol{\lambda}}^T)\Big)\hat{\mathbf{u}}\cdot\mathbf{n}\,d\mathbf{s} \\ &+ \int_{\Gamma_b} \Big(-\mathrm{I}\tilde{\alpha} + \nu(\nabla\tilde{\mathbf{u}} + \nabla\tilde{\mathbf{u}}^T) + \nu(\nabla\tilde{\boldsymbol{\lambda}} + \nabla\tilde{\boldsymbol{\lambda}}^T)\Big) : \nabla\mathbf{u}\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s} \\ &+ \int_{\Gamma_b} \nabla\cdot\Big(-\mathrm{I}\tilde{\alpha} + \nu(\nabla\tilde{\mathbf{u}} + \nabla\tilde{\mathbf{u}}^T) + \nu(\nabla\tilde{\boldsymbol{\lambda}} + \nabla\tilde{\boldsymbol{\lambda}}^T)\Big)\cdot\mathbf{u}\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s}\end{aligned} \tag{3.11}$$

Setting (3.9) and (3.10) to zero leads to the following strong form for the incremental state equation for $(\hat{\mathbf{u}},\hat{p})$:

$$-\nabla\cdot\big(-\mathrm{I}\hat{p} + \nu(\nabla\hat{\mathbf{u}} + \nabla\hat{\mathbf{u}}^T)\big) + \mathbf{u}\cdot\nabla\hat{\mathbf{u}} + \hat{\mathbf{u}}\cdot\nabla\mathbf{u} = \mathbf{0} \text{ in } \Omega,$$

$$
\begin{aligned}
\nabla\cdot\hat{\mathbf{u}} &= 0 \ \text{ in } \Omega, && (3.12)\\
\hat{\mathbf{u}}+(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\mathbf{n}\mathbf{W}\cdot\mathbf{n} &= \mathbf{0} \ \text{ on } \Gamma_b.
\end{aligned}
$$

Additionally, $\hat{\mathbf{u}}$ vanishes on Γ_t and is periodic on Γ_p. Next, the incremental adjoint equations are obtained from the variations of $\delta_u\mathscr{L}$ and $\delta_p\mathscr{L}$. From (3.4a), we obtain for all $(\tilde{\mathbf{u}},\tilde{p})\in\tilde{\mathbf{H}}^1_0(\Omega)\times L^2(\Omega)$:

$$
\begin{aligned}
\delta^2_{\mathbf{u}\bullet}\mathscr{D}(\Gamma_b,\mathbf{W}) = &\int_\Omega \frac{\nu}{2}(\nabla\hat{\mathbf{u}}+\nabla\hat{\mathbf{u}}^T):(\nabla\tilde{\mathbf{u}}+\nabla\tilde{\mathbf{u}}^T)+\frac{\nu}{2}(\nabla\hat{\boldsymbol{\lambda}}+\nabla\hat{\boldsymbol{\lambda}}^T):(\nabla\tilde{\mathbf{u}}+\nabla\tilde{\mathbf{u}}^T)\,d\mathbf{x}\\
&-\int_\Omega \hat{\alpha}\nabla\cdot\tilde{\mathbf{u}}\,d\mathbf{x}+\int_\Omega\Big((\tilde{\mathbf{u}}\cdot\nabla\hat{\mathbf{u}}+\hat{\mathbf{u}}\cdot\nabla\tilde{\mathbf{u}})\cdot\boldsymbol{\lambda}+(\tilde{\mathbf{u}}\cdot\nabla\mathbf{u}+\mathbf{u}\cdot\nabla\tilde{\mathbf{u}})\cdot\hat{\boldsymbol{\lambda}}\Big)\,d\mathbf{x}\\
&-\int_{\Gamma_b}\Big(-\mathrm{I}\,\hat{\alpha}+(\nabla\hat{\mathbf{u}}+\nabla\hat{\mathbf{u}}^T)+(\nabla\hat{\boldsymbol{\lambda}}+\nabla\hat{\boldsymbol{\lambda}}^T)\Big)\mathbf{n}\cdot\tilde{\mathbf{u}}\,d\mathbf{s}\\
&+\int_{\Gamma_b}\frac{\nu}{2}(\nabla\mathbf{u}+\nabla\mathbf{u}^T+\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T):(\nabla\tilde{\mathbf{u}}+\nabla\tilde{\mathbf{u}}^T)\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s}\\
&-\int_{\Gamma_b}\alpha\nabla\cdot\tilde{\mathbf{u}}\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s}+\int_{\Gamma_b}(\tilde{\mathbf{u}}\cdot\nabla\mathbf{u}+\mathbf{u}\cdot\nabla\tilde{\mathbf{u}})\cdot\boldsymbol{\lambda}\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s} \qquad (3.13)\\
&-\int_{\Gamma_b}\nabla\cdot\big(\boldsymbol{\lambda}\cdot\nu(\nabla\tilde{\mathbf{u}}+\nabla\tilde{\mathbf{u}}^T)\big)\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s}-\int_{\Gamma_b}\hat{\boldsymbol{\lambda}}\cdot\nu(\nabla\tilde{\mathbf{u}}+\nabla\tilde{\mathbf{u}}^T)\mathbf{n}\,d\mathbf{s}\\
&-\int_{\Gamma_b}\nabla\cdot\Big(\Big(-\mathrm{I}\,\alpha+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)+\nu(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T)\Big)\tilde{\mathbf{u}}\Big)\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s}.
\end{aligned}
$$

For all $\tilde{p}\in L^2(\Omega)$,

$$
\begin{aligned}
\delta^2_{p\bullet}\mathscr{D}(\Gamma_b,\mathbf{W}) &= \int_\Omega(-\nabla\cdot\hat{\boldsymbol{\lambda}})\tilde{p}\,d\boldsymbol{x}+\int_{\Gamma_b}\tilde{p}\hat{\boldsymbol{\lambda}}\cdot\mathbf{n}\,d\mathbf{s}+\int_{\Gamma_b}(-\nabla\cdot\boldsymbol{\lambda})\tilde{p}\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s}\\
&\quad+\int_{\Gamma_b}\nabla\cdot(\tilde{p}\boldsymbol{\lambda})\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s},\\
&= \int_\Omega(-\nabla\cdot\hat{\boldsymbol{\lambda}})\tilde{p}\,d\boldsymbol{x}+\int_{\Gamma_b}\tilde{p}\hat{\boldsymbol{\lambda}}\cdot\mathbf{n}\,d\mathbf{s}+\int_{\Gamma_b}\nabla\tilde{p}\cdot\boldsymbol{\lambda}\mathbf{W}\cdot\mathbf{n}\,d\mathbf{s} \qquad (3.14)\\
&= \int_\Omega(-\nabla\cdot\hat{\boldsymbol{\lambda}})\tilde{p}\,d\boldsymbol{x}.
\end{aligned}
$$

The variational forms (3.13), (3.14) are equivalent to the following strong form of the incremental adjoint equations:

$$
\begin{aligned}
-\nabla\cdot\Big(-\mathrm{I}\,\hat{\alpha}+\nu(\nabla\hat{\mathbf{u}}+\nabla\hat{\mathbf{u}}^T)+\nu(\nabla\hat{\boldsymbol{\lambda}}+\nabla\hat{\boldsymbol{\lambda}}^T)\Big) &= \\
-\nabla\hat{\mathbf{u}}^T\boldsymbol{\lambda}-\nabla\mathbf{u}^T\hat{\boldsymbol{\lambda}}+\nabla\boldsymbol{\lambda}\hat{\mathbf{u}}+\nabla\hat{\boldsymbol{\lambda}}\mathbf{u} & \qquad \text{in } \Omega, \qquad (3.15a)\\
\nabla\cdot\hat{\boldsymbol{\lambda}} = 0 & \qquad \text{in } \Omega, \qquad (3.15b)\\
\hat{\boldsymbol{\lambda}}+(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T)\mathbf{n}(\mathbf{W}\cdot\mathbf{n}) = 0 & \qquad \text{on } \Gamma_b. \qquad (3.15c)
\end{aligned}
$$

Additionally, $\hat{\boldsymbol{\lambda}} = 0$ on Γ_t and is periodic on Γ_p. Defining $\mathbf{V}$ and $\mathbf{W}$ as shape perturbations in the velocity method and using (2.4) we are able to obtain a simplified form for the evaluation of the shape Hessian in directions $(\mathbf{V},\mathbf{W})$ since the residuals of the state and adjoint momentum and mass equations on the boundary vanish:

$$
\begin{aligned}
d^2\mathscr{D}(\Gamma_b;\mathbf{V},\mathbf{W}) = \int_{\Gamma_b}\Big(-\frac{\nu}{2}(\nabla\mathbf{u}+\nabla\mathbf{u}^T):(\nabla\hat{\mathbf{u}}+\nabla\hat{\mathbf{u}}^T)-\frac{\nu}{2}(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T):\\
(\nabla\hat{\mathbf{u}}+\nabla\hat{\mathbf{u}}^T)-\frac{\nu}{2}(\nabla\hat{\boldsymbol{\lambda}}+\nabla\hat{\boldsymbol{\lambda}}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\Big)\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s} \qquad (3.16)\\
+\int_{\Gamma_b}\nabla\cdot\Big(-\frac{\nu}{4}(\nabla\mathbf{u}+\nabla\mathbf{u}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\mathbf{V}\Big)(\mathbf{W}\cdot\mathbf{n})\,d\mathbf{s}\\
+\int_{\Gamma_b}\nabla\cdot\Big(-\frac{\nu}{2}(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T):(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\mathbf{V}\Big)(\mathbf{W}\cdot\mathbf{n})\,d\mathbf{s}.
\end{aligned}
$$

To obtain this final simplified form of the shape Hessian, we have used the fact that since Γ_p and Γ_t are fixed boundaries (i.e., not subject to perturbations), the normal components $\mathbf{V}\cdot\mathbf{n}$ and $\mathbf{W}\cdot\mathbf{n}$ of the shape perturbations have to vanish on these boundaries. Thus, one can rewrite the integral over Γ_b in (3.7) as an integral over $\partial\Omega$ and apply (2.4).

We remark that compared to the incremental state equation in [64][1], the boundary condition in (3.12) contains the additional term $\nabla \mathbf{u}^T \mathbf{n}$ and $\nabla \boldsymbol{\lambda}^T \mathbf{n}$, respectively. However, using the homogeneous boundary condition for $\mathbf{u}$ and $\boldsymbol{\lambda}$ on Γ_b, as well as the divergence-free condition, one can argue that $\nabla \mathbf{u}^T \mathbf{n} = 0$ and $\nabla \boldsymbol{\lambda}^T \mathbf{n} = 0$. This argument is similar to the proof of (3.8). Summarizing, we have the following theorem.

Theorem 3.3.1. *The application of the shape Hessian operator at* Ω *to directions* $(\mathbf{V}, \mathbf{W})$ *is given by* (3.16)*, where* $(\mathbf{u}, p)$ *and* $(\boldsymbol{\lambda}, \alpha)$ *denote the solution to the state and adjoint equations, respectively, and* $(\hat{\mathbf{u}}, \hat{p})$ *and* $(\hat{\boldsymbol{\lambda}}, \hat{\alpha})$ *the solution to the incremental state and incremental adjoint equations* (3.12) *and* (3.15)*, respectively.*

---

[1] The incremental state equation is referred to as local shape derivative in [64], where it is used to derive the shape gradient.

Chapter 4

Steady Couette and Poiseuille flows

In this chapter, the shape gradient and shape Hessian given in Theorems 3.2.1 and 3.3.1 are evaluated for the special case of a flat boundary. In this case, analytical Hessians are found for Stokes flow while only numerical Hessians are available for Navier-Stokes flow.

It will be shown below that the flat boundary is a stationary point of the roughness functional $\mathscr{D}$, that is, the shape gradient vanishes. Hence, the behavior of $\mathscr{D}$ around the flat boundary is dominated by its second-order shape derivatives as illustrated by the Taylor expansion:

$$\mathscr{D}(\Gamma_b) \approx \mathscr{D}(\Gamma_b^0) + d\mathscr{D}(\Gamma_b^0, \mathbf{V}) + \frac{1}{2} d^2 \mathscr{D}(\Gamma_b^0; \mathbf{V}, \mathbf{V}), \tag{4.1}$$

where Γ_b is the perturbed boundary given by:

$$\Gamma_b = \{\mathbf{x} : \mathbf{x} = \mathbf{X} + \mathbf{V}(\mathbf{X}), \text{for } \mathbf{X} \in \Gamma_b^0\}.$$

Since this analysis is for a domain with fixed volume, the admissible boundary perturbations $\mathbf{V}(\mathbf{X})$ must satisfy

$$\int_{\Gamma_b} \mathbf{V} \cdot \mathbf{n} \, d\mathbf{s} = 0. \tag{4.2}$$

In the sequel, we restrict ourselves to considering only boundary perturbations in the wall normal direction. This assumption limits the roughness topographies we permit, but enables the use of Fourier modes as a parametrization. However, it excludes certain topographies, for instance those that cannot be written as a single-valued function of x and y.

Due to the periodicity of the boundary perturbations in both x and y, it will be convenient to expand the boundary perturbations in a Fourier basis. To this end, the normalized one-dimensional Fourier basis in the streamwise (x) direction is defined as:

$$\psi_j^x = \frac{1}{\sqrt{L_x}} \begin{cases} 1 & \text{if } j = 0, \\ \sqrt{2}\sin(k_x(j)x) & \text{if } j > 0, \\ \sqrt{2}\cos(k_x(j)x) & \text{if } j < 0. \end{cases}$$

where L_x is the length of the computational domain in the x direction and $k_x(j) = \frac{2\pi|j|}{L_x}$ is the wavenumber. Fourier basis functions ψ_j^y in the spanwise (y) direction are defined similarly, with x replaced by y, k_x replaced by $k_y(j) = \frac{2\pi|j|}{L_y}$ and L_x replaced by L_y (the domain size in the y direction). A basis for the boundary perturbations is then given by $\mathbf{V}_{i,j} = (0, 0, \psi_{i,j})^T$ with $\psi_{i,j} = \psi_i^x \psi_j^y$, where because the mean distance between the top and bottom walls is constrained to remain constant h, the function ψ_{00} is excluded from the basis of allowed shape perturbations. Note that for these shape perturbations $\mathbf{V}_{i,j}$, the condition (2.5) is satisfied, and thus the shape Hessian will be symmetric. Moreover, for $\mathbf{V}_{i,j}$ defined above, we have $\nabla \cdot \mathbf{V}_{i,j} = 0$. Thus using the regularity of Ω, the boundary data and the forcing $\mathbf{f}$, we can define the following function $h(\mathbf{W})$ on Γ_b:

$$\begin{aligned} h(\mathbf{W}) = \big(&-\frac{\nu}{2}(\nabla\mathbf{u} + \nabla\mathbf{u}^T) : (\nabla\hat{\mathbf{u}} + \nabla\hat{\mathbf{u}}^T) - \frac{\nu}{2}(\nabla\boldsymbol{\lambda} + \nabla\boldsymbol{\lambda}^T) : (\nabla\hat{\mathbf{u}} + \nabla\hat{\mathbf{u}}^T) \\ &- \frac{\nu}{2}(\nabla\hat{\boldsymbol{\lambda}} + \nabla\hat{\boldsymbol{\lambda}}^T) : (\nabla\mathbf{u} + \nabla\mathbf{u}^T)\big)\mathbf{n} \\ &+ \nabla\left(-\frac{\nu}{4}(\nabla\mathbf{u} + \nabla\mathbf{u}^T + \nabla\boldsymbol{\lambda} + \nabla\boldsymbol{\lambda}^T) : (\nabla\mathbf{u} + \nabla\mathbf{u}^T)\right)(\mathbf{W} \cdot \mathbf{n}). \end{aligned} \tag{4.3}$$

Then, the application of the shape Hessian (3.16) to shape directions $(\mathbf{V},\mathbf{W})$ can be expressed as L^2-inner product:

$$d^2\mathscr{D}(\Gamma_b;\mathbf{V},\mathbf{W}) = \int_{\Gamma_b} h(\mathbf{W})\cdot\mathbf{V}\,d\mathbf{s}.$$

Moreover, due to the regularity assumptions, $h(\mathbf{W})$ is Lipschtiz continuous on Γ_b. Thus, the Hessian operator maps a Lipschitz continuous function to a Lipschtiz continuous function. The characterization of the shape Hessian for the flat boundary is greatly simplified by the following lemma.

Lemma 4.0.2. *The Fourier basis functions are eigenfunctions of the Navier-Stokes shape Hessian operator at the flat boundary* Γ_b *provided the forcing* $\mathbf{f}$ *and the boundary conditions on* Γ_t *are homogeneous (i.e., independent of* x *and* y*). To be precise, in the three-dimensional case the eigenfunctions are given by* $\mathbf{V}_{i,j}\cdot\mathbf{n} = \psi_{i,j}$*, with* $(i,j)\in\mathbb{Z}^2$*, excluding* $i=j=0$*.*

Proof. Homogeneity in x and y of $\mathbf{f}$ and the boundary conditions imply that the flow is homogeneous. Thus, the velocity and any function of the velocity, in particular, the shape Hessian, are homogeneous. We define the operator H that maps the normal boundary perturbations $\psi_{i,j}$ for the flat-boundary Hessian to $h(\mathbf{V}_{i,j})^T\mathbf{n}$, with $h(\mathbf{V})$ as defined in (4.3). Thus, for $i,j,k,l\in\mathbb{Z}$,

$$\int_{\Gamma_b} (\mathrm{H}(\psi_{i,j}))\,\psi_{k,l} = d^2\mathscr{D}(\Gamma_b^0,\mathbf{V}_{i,j},\mathbf{V}_{k,l}).$$

The homogeneity of the shape Hessian implies that H commutes with translations (H is said to be a *translation-invariant operator*); therefore, for any $\psi_{i,j}$,

$$\tau_{x_0,y_0}(\mathrm{H}\,\psi_{i,j}) = \mathrm{H}(\tau_{x_0,y_0}\psi_{i,j}),$$

where τ_{x_0,y_0} is the translation-by-(x_0,y_0) operator, that is, $\tau_{x_0,y_0}f(x,y) = f(x-x_0,y-y_0)$ for a function f. Note that the Fourier basis functions defined above satisfy

$$(\tau_{-x_0}\psi_i^x)(x) = \psi_i^x(x+x_0) = \begin{cases} \psi_i^x(x)\psi_{-i}^x(x_0) + \psi_{-i}^x(x)\psi_i^x(x_0) & \text{for } i>0, \\ \psi_i^x(x) & \text{for } i=0, \\ \psi_i^x(x)\psi_i^x(x_0) - \psi_{-i}^x(x)\psi_{-i}^x(x_0) & \text{for } i<0. \end{cases} \tag{4.4}$$

In the remainder of the proof, $i,j>0$ is imposed, and negative subscripts on ψ are indicated explicitly. With this convention, (4.4) implies

$$\begin{aligned} \tau_{-x_0,-y_0}\psi_{i,j} = {}& \psi_{-i,-j}(x_0,y_0)\psi_{i,j} + \psi_{i,j}(x_0,y_0)\psi_{-i,-j} \\ &+ \psi_{-i,j}(x_0,y_0)\psi_{i,-j} + \psi_{i,-j}(x_0,y_0)\psi_{-i,j}, \end{aligned} \tag{4.5}$$

and similar relations hold for $\psi_{\pm i,\pm j}$, $\psi_{0,\pm i}$ and $\psi_{\pm j,0}$. Using the translation invariance of H, the fact that $\mathrm{H}\,\psi_{i,j}$ can be understood as a pointwise defined function and (4.5) result in

$$\begin{aligned} (\mathrm{H}\,\psi_{i,j})(x_0,y_0) &= (\tau_{-x_0,-y_0}\mathrm{H}\,\psi_{i,j})(0,0) = (\mathrm{H}(\tau_{-x_0,-y_0}\psi_{i,j}))(0,0) \\ &= \psi_{i,j}(x_0,y_0)(\mathrm{H}\,\psi_{-i,-j})(0,0) + \psi_{-i,-j}(x_0,y_0)(\mathrm{H}\,\psi_{i,j})(0,0) \\ &\quad + \psi_{-i,j}(x_0,y_0)(\mathrm{H}\,\psi_{i,-j})(0,0) + \psi_{i,-j}(x_0,y_0)(\mathrm{H}\,\psi_{-i,j})(0,0). \end{aligned} \tag{4.6}$$

Analogously, for $(i,-j)$ one obtains that

$$\begin{aligned} (\mathrm{H}\,\psi_{i,-j})(x_0,y_0) = {}& \psi_{-i,-j}(x_0,y_0)(\mathrm{H}\,\psi_{i,-j})(0,0) - \psi_{i,j}(x_0,y_0)(\mathrm{H}\,\psi_{-i,j})(0,0) \\ &- \psi_{-i,j}(x_0,y_0)(\mathrm{H}\,\psi_{i,j})(0,0) + \psi_{i,-j}(x_0,y_0)(\mathrm{H}\,\psi_{-i,-j})(0,0). \end{aligned} \tag{4.7}$$

Similar expressions hold for $(-i,-j)$, $(-i,j)$ and for $(i,0)$, $(0,j)$. Using the fact that the Hessian and thus H is a symmetric operator, the orthonormality of the

Fourier modes together with (4.6) and (4.7) implies

$$(\mathrm{H}\,\psi_{-i,j})(0,0) = \int_{\Gamma_b} \mathrm{H}\,\psi_{i,j}\psi_{i,-j}\,d\mathbf{s} = \int_{\Gamma_b} \mathrm{H}\,\psi_{i,-j}\psi_{i,j}\,d\mathbf{s} = -(\mathrm{H}\,\psi_{-i,j})(0,0),$$

which shows that $(\mathrm{H}\,\psi_{-i,j})(0,0) = 0$. Similarly, it follows that $(\mathrm{H}\,\psi_{i,-j})(0,0) = (\mathrm{H}\,\psi_{i,j})(0,0) = 0$, so that $(\mathrm{H}\,\psi_{i,j})(0,0)$ is non-zero only if both subscripts are non-positive. With this result, only one term in (4.6), (4.7) and the analogous expressions survives, yielding

$$\mathrm{H}\,\psi_{\pm i,\pm j} = (\mathrm{H}\,\psi_{-i,-j})(0,0)\,\psi_{\pm i,\pm j}. \tag{4.8}$$

A brief computation shows that the eigenvalue equation (4.8) also holds for $i = 0$ or $j = 0$, completing the proof. □

Note that Lemma (4.0.2) can be seen as a particular case of a general result on translation-invariant operators. In particular, in [30] it is shown that translation-invariant operators between proper L_p-spaces are convolutions, which are known to have Fourier functions as eigenfunctions. Note that the homogeneity assumption for the flow in Lemma 4.0.2 is only satisfied for homogeneous boundary conditions, volume force and for flat Γ_b. If these conditions are not satisfied, the eigenfunctions are not known *a priori*. Also, observe in (4.8) that the eigenvalues for $\psi_{i,j}$ and $\psi_{i,-j}$ are the same, which is also observed in the examples below.

In the first three examples, the influence of roughness on Stokes flow is considered. Stokes flow is the limit case of Navier-Stokes flow as the Reynolds number tends to zero, which results in the nonlinear term $\mathbf{u} \cdot \nabla \mathbf{u}$ vanishing. Thus, for Stokes flow, the nonlinear term and its linearizations vanish in the state, adjoint, incremental state and incremental adjoint equations, while the expressions for the gradient (3.7) and Hessian (3.16) remain unchanged. Three of the examples below are posed in two-dimensional domains Ω. Thus, the deformation velocity fields simplify to $\mathbf{V}_j = (0, \psi_j^x)$ and a result analogous to Lemma 4.0.2 holds.

4.1 Two-dimensional Stokes Couette flow

The first example is a two-dimensional Stokes Couette flow, that is, the flow is driven by the top boundary moving at constant velocity $\mathbf{u}_0 = (U_t, 0)^T$. Since the bottom boundary Γ_b is flat, the state and adjoint systems can be solved analytically on the domain $(0, L_x) \times (0, h)$, to obtain $\mathbf{u} = U_t/h(y, 0)^T$ and $\boldsymbol{\lambda} = (0, 0)^T$ with h being the distance between the two walls. Moreover, the pressure p and adjoint pressure α are constant. Thus, for the shape gradient in any admissible direction $\mathbf{V}$ we obtain:

$$d\mathscr{D}(\Gamma_b; \mathbf{V}) = -\frac{\nu U_t^2}{2h^2} \int_{\Gamma_b} \mathbf{V} \cdot \mathbf{n}\, ds = 0.$$

The shape derivative is thus zero and the flat boundary is a stationary point of the roughness functional. Since the Fourier functions are the eigenfunctions of the shape Hessian, the Hessian can be completely determined by evaluating it for shape perturbations in directions $\mathbf{V}_l$ and $\mathbf{W}_j$. The solutions to the incremental state and adjoint equations (3.12) and (3.15) can be calculated explicitly, and, letting $k = k_x(j)$, are given by:

$$\hat{\mathbf{u}} = \begin{pmatrix} -[c_1 e^{ky} - c_2 e^{-ky} + c_3(\frac{1}{k} + y)e^{ky} + c_4(\frac{1}{k} - y)e^{-ky}]\psi_j^x(x) \\ (c_1 e^{ky} + c_2 e^{-ky} + c_3 y e^{ky} + c_4 y e^{-ky})(\psi_j^x)'(x) \end{pmatrix}, \quad \hat{\boldsymbol{\lambda}} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

where

$$c_1 = \frac{hU_t k^2}{2h^2k^2 + 1 - \cosh(2kh)}, \qquad c_2 = -\frac{hU_t k^2}{2h^2k^2 + 1 - \cosh(2kh)},$$
$$c_3 = \frac{U_t k}{h}\frac{e^{-kh}[-\sinh(kh) + khe^{kh}]}{2h^2k^2 + 1 - \cosh(2kh)}, \qquad c_4 = -\frac{U_t k}{h}\frac{e^{kh}[-\sinh(kh) + hke^{-kh}]}{2h^2k^2 + 1 - \cosh(2kh)}.$$

Note that the incremental solution $\hat{\mathbf{u}}$ exponentially decays to zero in wall normal direction. Its scale height, which is the size of the layer in which the incremental solution is significant, is at the order of the roughness wavelength (i.e., $2\pi/k$). Evaluating (3.16), the Hessian is found to be

$$d^2\mathscr{D}(\Gamma_b;\mathbf{V},\mathbf{W}) = \frac{S^2\nu L_x}{h}\gamma_c(k)\delta_{jl} \quad \text{with } \gamma_c(k) = \frac{2kh(2kh - \sinh(2kh))}{1 + 2k^2h^2 - \cosh(2kh)} > 0,$$

where δ_{jl} is the Kronecker delta and $S = U_t/h$. The diagonal structure of the Hessian operator is a consequence of the Fourier modes being eigenfunctions and $\gamma_c(k)$ is seen to be the scaled (i.e., dimensionless) eigenvalue associated with Fourier mode with wavenumber k. The eigenvalues are plotted in Figure 4.1. Note that for large k,

$$\gamma_c(k) \approx 2kh, \qquad \text{where} \quad k = 2\pi j/L_x.$$

4.2 Two-dimensional Stokes Poiseuille flow

Next, consider two-dimensional Stokes Poiseuille flow (channel flow), in which no-flow conditions $\mathbf{u}_0 = \mathbf{0}$ hold on Γ_t and the body force $\mathbf{f}$ is specified such that the total mass flux through the domain Ω is fixed, so that

$$\int_\Omega u_1\, d\mathbf{x} = c, \qquad \int_\Omega u_2\, d\mathbf{x} = 0,$$

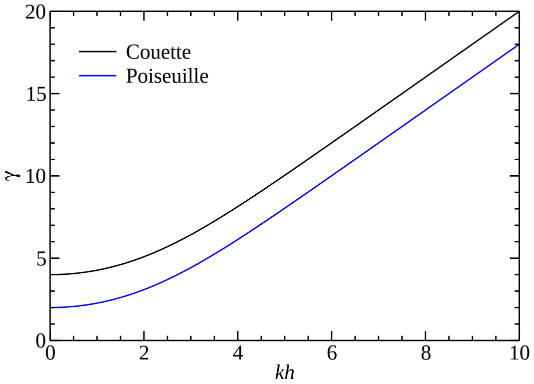

Figure 4.1: Eigenvalues of shape Hessian for two-dimensional Stokes flows for various frequencies kh of the shape perturbation.

which implies that $\mathbf{f} = (\frac{12c}{\nu} L_x h^3, 0)^T$. In this case, the solutions to the state and the adjoint equations are

$$\mathbf{u} = \frac{6c}{L_x h^3} \begin{pmatrix} -y^2 + hy \\ 0 \end{pmatrix}, \qquad \boldsymbol{\lambda} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

with constant pressure and adjoint pressure p and α, respectively. The shape gradient becomes

$$d\mathscr{D}(\Gamma_b; \mathbf{V}) = -\frac{18c^2\nu}{L_x^2 h^4} \int_{\Gamma_b} \mathbf{V} \cdot \mathbf{n}\, ds = 0 \text{ for all admissible } \mathbf{V}.$$

Thus, again the flat boundary is a stationary point of $\mathscr{D}$. Also, for Poiseuille flow, the solutions to the incremental equations (3.12) and (3.15) can be derived analytically, and the application of the shape Hessian to $\mathbf{V}_j$, $\mathbf{W}_l$ becomes, with $k = k_x(j)$

$$d^2\mathscr{D}(\Gamma_b; \mathbf{V}, \mathbf{W}) = \frac{S^2 \nu L_x}{h} \gamma_p(k) \delta_{jl} \quad \text{with } \gamma_p(k) = \frac{2kh(2kh - \sinh(2kh))}{1 + 2k^2h^2 - \cosh(2kh)} - 2,$$

where $S = 6c/L_x h^2$ is the gradient of the state solution on Γ_b. The eigenvalues for the Poiseuille case are also shown in Figure 4.1. For large k,

$$\gamma_p(k) \approx 2kh - 2.$$

4.3 Three-dimensional Couette flow

The two-dimensional results from Section 4.1 can be generalized to three-dimensional Couette Stokes flow with two-dimensional roughness. The flow is driven by the boundary condition $\mathbf{u}_0 = (U_t, 0, 0)^T$ on Γ_t. As in the two-dimensional case, solutions to the state and adjoint equations are found as

$$\mathbf{u} = \frac{U_t}{h}\begin{pmatrix} z \\ 0 \\ 0 \end{pmatrix}, \qquad \boldsymbol{\lambda} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \qquad p = \text{constant}, \qquad \alpha = \text{constant}.$$

so that, as before, the shape gradient vanishes, making the flat wall a stationary point of the roughness functional $\mathscr{D}$. Solving the incremental equations as in the two-dimensional case yields for the application of the Hessian to directions $\mathbf{V}_{ij}$ and $\mathbf{W}_{kl}$:

$$d^2\mathscr{D}(\Gamma_b; \mathbf{V}, \mathbf{W}) = \frac{S^2 \nu L_x L_y}{h} \gamma(k_x, k_y) \delta_{ik} \delta_{jl},$$

where, again the i- and j-dependence of $k_x(i)$ and $k_y(j)$ is not shown, and

$$\gamma(k_x, k_y) = \left(\frac{k_x}{k}\right)^2 \frac{2kh(2kh - \sinh(2kh))}{1 + 2k^2h^2 - \cosh(2kh)} + \left(\frac{k_y}{k}\right)^2 \frac{kh\cosh(kh)}{\sinh(kh)}$$

with $k = \sqrt{k_x^2 + k_y^2}, S = U_t/h$. Here, $\gamma(k_x, k_y)$ is the eigenvalue associated with wavenumber (k_x, k_y). For large k,one obtains $\gamma(k_x, k_y) \approx kh(1 + (k_x/k)^2)$ and, for a constant ration k_x/k_y, γ_{k_x,k_y} increases linearly with k. Moreover, for fixed k, γ_{k_x,k_y} is maximal in the case $k_y = 0$ and minimal for $k_x = 0$.

Note that it is possible to obtain analytical solutions in the above examples because the incremental equations for Stokes flow can be solved analytically. To verify the above findings, the state equation was solved numerically on perturbed domains and finite differences were used to compute the shape gradient and shape Hessian. The difference between the resulting numerically computed and the analytical eigenvalues was found to be at the order of the discretization error. More information concerning the numerical solution is provided in the Navier-Stokes example described below, where analytical solutions are not available.

4.4 Navier-Stokes Couette flow

Finally, consider two-dimensional Couette flow governed by the Navier-Stokes equations. The state and adjoint equations at the flat boundary have the same solutions as for Stokes flow, namely:

$$\mathbf{u} = \frac{U_t}{h}\begin{pmatrix} y \\ 0 \end{pmatrix}, \qquad \boldsymbol{\lambda} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad p = \text{constant}, \qquad \alpha = \text{constant}. \tag{4.9}$$

Thus, also for Navier-Stokes flow, the shape gradient vanishes at the flat boundary. Since analytical solutions to the incremental solutions are not available, the incremental equations must be solved numerically. A numerical Hessian can then be found by evaluating the expression (3.16) using the approximate solutions of the incremental equations. As proven in Lemma 4.0.2, the Fourier modes are the eigenfunctions of the shape Hessian for the flat boundary. This implies that, in the Fourier basis, the shape Hessian is diagonal and hence, a single pair of incremental state and incremental adjoint solves is sufficient to find all the eigenvalues. We choose **W** as a sum of eigenfunctions (i.e., Fourier modes), solve the incremental equations and use individual Fourier modes for **V** in (3.16) to compute the diagonal elements of the shape Hessian.

To obtain numerical solutions to the incremental state and incremental adjoint equations, the computational domain, Ω $(= [0,L_x] \times [0,h] \times [0,L_z])$, is discretized with hexahedral Q_4–Q_2 finite elements (i.e. biquartic polynomials represent the components of the incremental velocities $\hat{\mathbf{u}}, \hat{\boldsymbol{\lambda}}$, and biquadratic elements represent the incremental pressures $\hat{p}$, $\hat{q}$). The mesh is graded towards the bottom boundary to resolve boundary layer effects. To ensure accuracy of the numerical solution, empirical convergence studies were performed using a hierarchy of meshes, and an escalation of polynomial order of the finite element approximation. We found that a (graded) mesh with 100×100 elements gives highly accurate results for the Reynolds numbers considered.

The results of these computations show that the shape Hessian eigenvalues, and thus the sensitivity of the roughness functional to Fourier basis perturbations, increases with the Reynolds number based on h, i.e., $\mathrm{Re}_h = U_t h/\nu$. For modes for which $kh > 2\pi$, so that the scale height of the Stokes incremental solutions (which is order $2\pi/k$) is small compared to the fluid layer thickness h, the shape Hessian eigenvalues for different wave numbers and Reynolds numbers can be made to collapse on a single curve. In fact, the ratio of the Navier-Stokes shape Hessian eigenvalues γ_{NS} to those for Stokes flow γ_{S} depends only on the Reynolds number Re^*

$$\mathrm{Re}^* := \frac{U_t \lambda^2}{\nu h} = \mathrm{Re}_h \frac{\lambda^2}{h^2}, \tag{4.10}$$

where $\lambda = 2\pi/k$ is the wavelength of the roughness Fourier mode. The wavelength is also the scale height of the incremental solution, so Re^* is the Reynolds number based on the scale height λ and on the velocity $\lambda dU/dz = \lambda U_t/h$ at the scale height.

This collapse of the ratios between Stokes and Navier-Stokes eigenvalues when plotted against Re^* is shown in Figure 4.2. Additionally, the plot shows that Re^* determines the range in which the Stokes eigenvalues are a good approximation for the Navier Stokes eigenvalues and how they relate as Re^* increases.

Further insight into this behavior of the eigenvalues can be gained from scaling analysis. At distances from the wall greater than the scale height λ, the incremental solutions decay exponentially to zero. Thus the incremental solutions are insensitive to features of the state solution farther from the wall than $\sim\lambda$. If λ/h is sufficiently small, then the incremental solutions will not depend on U_t or h . Instead, the only property of the state solution on which the incrementals can depend is the velocity gradient $S = dU/dz = U_t/h$. The incremental solutions and therefore the Hessians can thus only depend on three dimensional parameters (S, λ, and ν), so by dimensional analysis, the non-dimensional Hessian eigenvalues ($\gamma_{\mathrm{NS}}/\gamma_{\mathrm{S}}$) depend only on the dimensionless parameter $S\lambda^2/\nu = \mathrm{Re}^*$. The same argument holds in other flows (e.g., Poiseuille flow), provided that the scale height of the incremental solution is small compared to flow features in the state solution.

The Reynolds number Re^* can also be interpreted in terms of the wall scaling commonly used in the analysis of wall-bounded turbulent flows, see e.g., [59]. In this scaling, the *wall or friction velocity* u_τ is defined by $u_\tau^2 = \nu dU/dz = \nu U_t/h$, and the *viscous length scale* δ_ν is defined as $\delta_\nu = \nu/u_\tau$. The Reynolds number (4.10) is then given by $\mathrm{Re}^* = (\lambda^+)^2$, where $\lambda^+ = \lambda/\delta_\nu$ is the wavelength (and thus also the scale height) normalized by the viscous length scale. Since $\lambda^+ = u_\tau\lambda/\nu$, it can also be understood as a local Reynolds number based on u_τ and λ, which

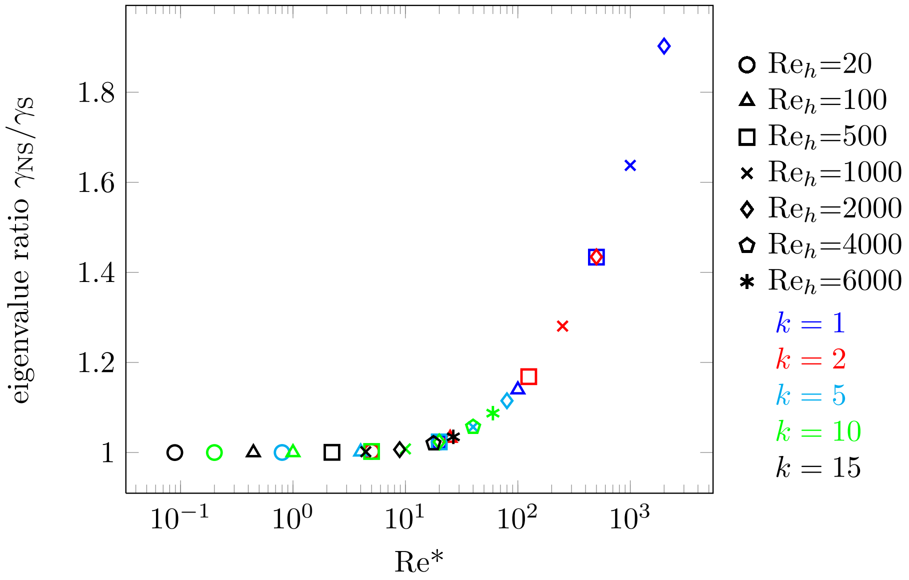

Figure 4.2: Ratio of shape Hessian eigenvalues of Navier-Stokes and Stokes flow for the case of Couette flow. The plot shows results for Re_h ranging between 20 and 6000.

measures the importance of inertial effects in the incremental solution, relative to viscous effects.

4.5 Flow predictions

The motivation of this analysis is to predict and understand how the characteristics of a roughness affect the fluid dynamics by using shape derivatives. Now, with the full knowledge of the shape gradient and shape Hessian for laminar flows, we can make predictions of the flow behavior for small roughnesses. In this section, we present numerical results of the prediction obtained from sensitivity analysis and study their accuracy by comparing them to the real values. Recall that for a 2D domain $\Omega(\Gamma_b)$ as given in Fig. 3.1, we consider the normalized dissipation,

$$\mathscr{D}(\Gamma_b) = \frac{\nu}{4|\Omega|} \int_{\Omega} (\nabla \mathbf{u} + \nabla \mathbf{u}^T) : (\nabla \mathbf{u} + \nabla \mathbf{u}^T)\, d\mathbf{x}, \tag{4.11}$$

where $|\Omega|$ denotes the area of Ω. A second-order Taylor expansion of (4.11) around the flat boundary

$$\mathscr{D}(\Gamma_b) \approx \mathscr{D}(\Gamma_b^0) + d\mathscr{D}(\Gamma_b^0, \mathbf{V}) + \frac{1}{2} d^2 \mathscr{D}(\Gamma_b^0; \mathbf{V}, \mathbf{V}), \tag{4.12}$$

gives:

$$\delta \mathscr{D} \approx S^2 \nu \sum_k (\alpha_k^2 + \beta_k^2) \gamma(k, \mathrm{Re}),$$

where $S = U_t/L_y$, L_x, L_y are the length and height of the channel, $\gamma(k, \mathrm{Re})$ the non-dimensional eigenvalue with wavenumber k and Reynolds number Re, $\mathbf{V} \cdot \mathbf{n}$ is the roughness topography (the shape perturbation) and α_k, β_k are the Fourier coefficients of its expansion, i.e.,

$$\mathbf{V} \cdot \mathbf{n} = \sum_k (\alpha_k \sin(kx) + \beta_k \cos(kx)).$$

First, let us find the relevant parameters for the variation of the dissipation $\delta \mathscr{D}$. For simplicity, we only consider a single Fourier mode ψ_k with wavenumber k and roughness height ε, namely, $\mathbf{V} \cdot \mathbf{n} = \varepsilon \psi_k$. Thus,

$$\delta \mathscr{D} \sim S^2 \nu \gamma_k \varepsilon^2.$$

Given that $u_\tau = \sqrt{\nu S}$,

$$\delta \mathscr{D} \sim S^2 \nu \gamma_k \varepsilon^2 = \frac{u_\tau^4 \nu}{\nu^2} \gamma_k \varepsilon^2 = u_\tau^3 (\frac{u_\tau}{\nu k})(\frac{\gamma_k}{k})(k\varepsilon)^2, \tag{4.13}$$

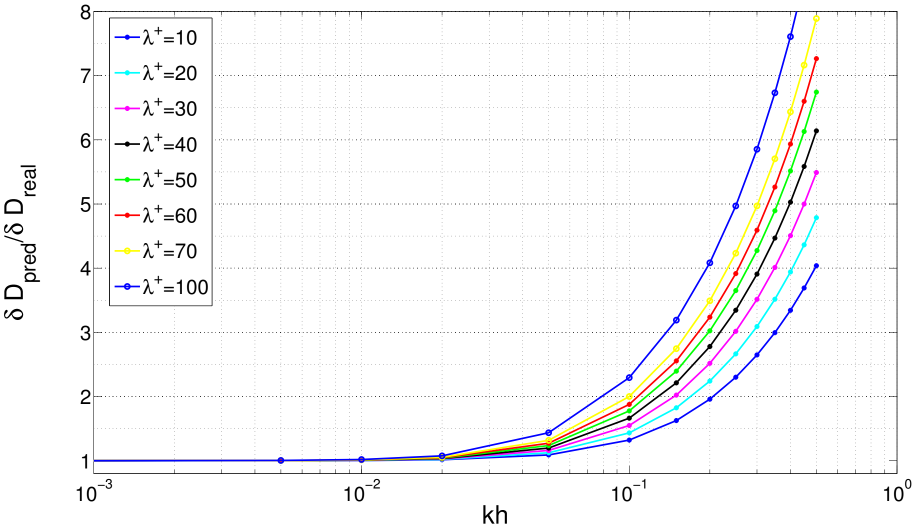

Figure 4.3: Comparison of predicted dissipation with real dissipation. The x-axis denotes the height of roughness, which is of order 0.1 ($k\varepsilon/L_y \approx 1\%$); the y-axis is the ratio of predicted variation with real variation of dissipation, i.e., $\delta\mathscr{D}_{pred}/\delta\mathscr{D}_{real}$; color denotes different λ^+ (thus kinematic viscosity ν).

where $u_\tau/(\nu k) = \lambda^+$, is a local Reynolds number based on u_τ and λ^+ as defined in Sec. 4.4, γ_k/k is the ratio of the eigenvalue for Navier-Stokes flow and eigenvalue for Stokes flow. As illustrated in Fig. 4.2, γ_k/k only depends on the characteristic Reynolds number $\mathrm{Re}^* = (\lambda^+)^2 = (u_\tau/(\nu k))^2$. Thus, $\delta\mathscr{D}$ can be fully characterized by Re^* (or equivalently λ^+) and $k\varepsilon$.

Now to study the accuracy of the prediction given using (4.12), we compute $\delta\mathscr{D}_{pred}$ following (4.12) and $\delta\mathscr{D}_{real} = \mathscr{D}(\Gamma_b) - \mathscr{D}(\Gamma_b^0)$ for different $k\varepsilon$ and $\lambda^+ = u_\tau/(\nu k)$. For computational convenience, we fix $L_x = 1, L_y = 10\pi$, where L_y is chosen large enough such that the influence from the boundary conditions on the top wall can be neglected; Since $k\varepsilon$ is the characteristic parameter rather than k or ε, we simply choose $k = 1$. The results of the comparison are shown in Fig. 4.3. It shows

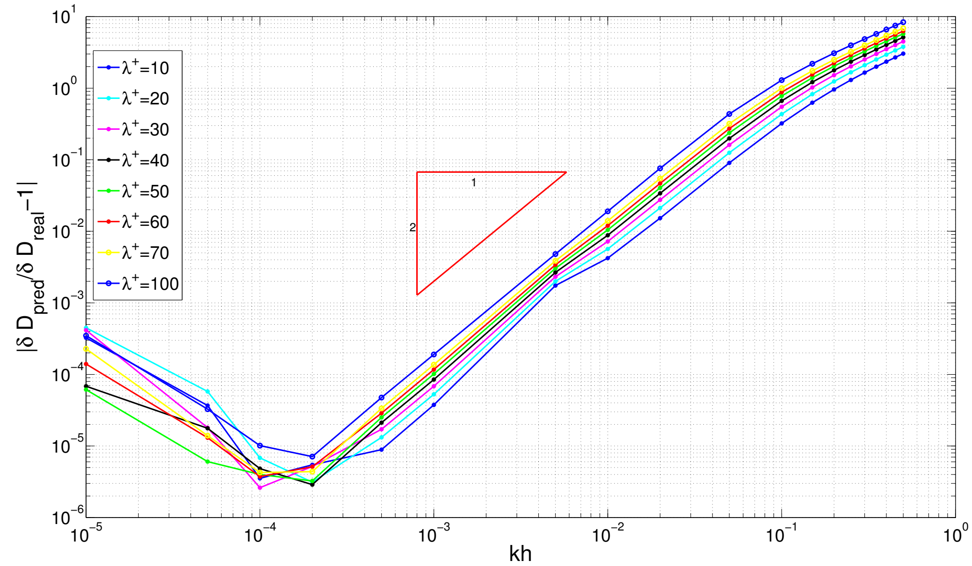

Figure 4.4: Behavior of the prediction errors of predicted dissipation compared with the real dissipation. The x-axis denotes the height of roughness; the y-axis is the errors of predicted variation; color denotes different λ^+; the triangle (in red) denotes the quadratic curve.

that for small roughness $k\varepsilon$, the ratio is close to one and the prediction describes the behavior of the drag functional accurately. However, when kh gets larger (for instance, $kh > 10^{-2}$), the prediction degrades. Moreover, the ratio diverges from 1 faster for large Reynolds number λ^+.

The behavior of the prediction errors is shown in Fig. 4.4. Here, we compute

$$\frac{\delta \mathscr{D}_{pred}}{\delta \mathscr{D}_{real}} - 1 = \frac{\delta \mathscr{D}_{pred} - \delta \mathscr{D}_{real}}{\delta \mathscr{D}_{real}}.$$

For small kh, the numerical errors in computing the real dissipation might be significant compared to δD_{real} (since $\delta D_{real} \sim \mathrm{o}((kh)^2)$), thus the computed prediction errors ($\delta \mathscr{D}_{pred}/\delta \mathscr{D}_{real} - 1$) can decrease in the region $kh \in [10^{-5}, 2\times 10^{-4}]$ as shown in Fig 4.4 because of the magnification of the numerical errors. For roughness with moderate height, the error grows quadratically with kh, which corresponds to the local second-order accuracy (for small kh).

4.6 Summary of the results for laminar flows

In this chapter, we showed results of shape derivatives evaluated at a flat boundary for the laminar flows. As expected, the analysis shows that the flat boundary is a stationary point of the roughness functional, so that the shape Hessian provides the lowest order description of the roughness effect. Furthermore, for a flat wall, the shape Hessian operator is translation invariant, so that its eigenfunctions are known *a priori* to be the Fourier functions. This allows the shape Hessian to be completely characterized by its eigenvalue spectrum, and greatly simplifies the determination of the Hessian.

The analysis reported here leads to the following observations regarding the sensitivity of the drag to roughness:

- The simple structure of the shape Hessian for Stokes flow allows the sensitivity of $\mathscr{D}$ to a general small-amplitude roughness to be determined easily from the roughness spectrum. Consider a two-dimensional channel with roughness

  $$\mathbf{V} \cdot \mathbf{n} = \sum_k (\alpha_k \sin(kx) + \beta_k \cos(kx))$$

  satisfying

  $$\sum_k \alpha_k^2 + \beta_k^2 < \infty, \text{ and } \sum_k (\alpha_k^2 + \beta_k^2) k < \infty. \tag{4.14}$$

Then, the drag increment due to this roughness is simply

$$\delta\mathscr{D} = \frac{L_x S^2 \nu}{2h} \sum_k (\alpha_k^2 + \beta_k^2)\gamma(k). \tag{4.15}$$

Due to the assumption (4.14) and the fact that $\gamma(k) \approx 2hk$ for large k, the sum in (4.15) converges. While only valid for small roughness heights, this is nonetheless a useful result as it provides a simple metric for the effect of roughness surface topography on the drag, and (4.1) (or (4.15)) applies to all roughnesses that are Lipschitz continuous and satisfy condition (4.14). The expression (4.15) also applies to the roughness analysis for Navier-Stokes flow with the Stokes Hessian eigenvalues $\gamma(k)$ replaced by the Navier-Stokes Hessian eigenvalues. The latter are well approximated by $\gamma(k)$ if Re^* (as defined in (4.10)) is small, see Figure 4.2.

- For all k, $\gamma_c(k) > 0$ and $\gamma_p(k) > 0$. Since all Hessian eigenvalues are positive, the flat boundary corresponds to a local minimum of the roughness functional. Moreover, for Stokes flow it is the global minimum, since the flat boundary is the only stationary point. This is due to the fact that $d\mathscr{D}(\Gamma_b; \mathbf{V})$ vanishes if and only if $\nabla\mathbf{u}$ is constant on Γ_b with $\boldsymbol{\lambda} = 0$ for Couette flow and $\boldsymbol{\lambda} = -\mathbf{u}$ for Poiseuille flow, and the flat boundary is the only shape for which this is true.

- In Stokes flow, for large wavenumbers ($kh > 2\pi$),

  $$\gamma_c(k) \approx 2kh, \qquad \gamma_p(k) \approx 2kh - 2,$$

  which shows that the eigenvalues increase linearly with wavenumber. Thus, the sensitivity of the roughness functional increases linearly with the wavenumber of the boundary perturbation. Provided $hk \gg 1$, that is, the height(h) of the channel is much larger than the scale height of the incremental solution,

$$\gamma_c(k) \approx \gamma_p(k) \approx 2kh.$$

In this case, the boundary conditions on the far outer flow do not influence the flow increment due to the roughness.

- The linear dependence of the eigenvalues with the wavenumber shows that the Hessian for Stokes flow is a pseudo-differential operator with order 1. This is also pointed out in [64], where the Hessian operator is considered as an input-to-output mapping in frequency space, which is motivated by the derivation of operator symbols.

- For small k, $\gamma(k)$ approaches the asymptotic value of 4 or 2 for Couette or Poiseuille flow, respectively. When $kh \ll 1$, then the scale height of the incremental solution is much larger than the fluid layer. The solution details in this case thus depend on the boundary conditions on the upper surface.

- For Navier-Stokes Couette flow, the flat boundary also corresponds to a stationary point of $\mathscr{D}$ (a minimum). The sensitivity of the roughness functional to Fourier basis perturbations increases with the Reynolds number. The ratio of the Navier-Stokes shape Hessian eigenvalues to those for Stokes flow depends only on the Reynolds number Re^* (as defined in (4.10)), which can be understood using scaling arguments.

- The scaling arguments mentioned above (see Section 4.4), also provide guidance for the applicability of the Navier-Stokes Couette flow analysis to roughness in a turbulent wall layer. For the analysis to apply, the scale height of the incremental solutions (of order λ) must not exceed the sublayer thickness, which is approximately $5\delta_\nu$ [59]. Therefore, of the results shown in figure 4.2, only those with $Re^* < 25$ ($\lambda^+ < 5$) are a valid approximation for turbulent flows.

However, note that roughness is of particular concern in turbulent flows. A particularly useful generalization of the steady Navier-Stokes shape Hessian analysis would thus be its application to turbulence. This would require generalization to the time-dependent Navier-Stokes state equation, whose analysis is followed later.

Chapter 5

Turbulent flows

We now extend previous analysis from laminar flows to the turbulent case. In the following analysis, all discussion is carried out under the constraints of the transient Navier-Stokes equations. A similar set of time-dependent adjoint and incremental equations will be derived and used to compute the corresponding shape derivatives. While for special cases such as those discussed in Chapter 4, i.e., steady Stokes or Navier-Stokes flows, analytical solutions exist or numerical solutions are easy to obtain, this is in general not the case for transient Navier-Stokes equations. Thus, a systematic study of the roughness functional must rely solely on numerical simulation. In this chapter, we derive the shape derivatives for time-dependent flows, then summarize the numerical methods used and describe the associated challenges.

5.1 Random nature of turbulence

Fluid flows can be divided into two different classes: laminar flows and turbulent flows. Laminar flows are generally smooth, quiet flows which vary in time only when the external forces or conditions are changing. Turbulent flows, which occur in the overwhelming majority of cases of technological interest, are flows accompanied by irregular fluctuations of all fluid mechanical quantities in time and space.

Although generally considered as deterministically characterized by the Navier-Stokes equations, turbulent flows are highly nonunique in practice. In any practical realization of a turbulent flow, there are unavoidable perturbations in initial and boundary conditions, and the detailed behavior of the flow is extremely sensitive to those small perturbations. Moreover, if turbulent flow is set up repeatedly under the same conditions, the exact values of these fields obtained will be different each time, i.e., the velocity fields can thus be considered to be *random* variables. Thus, the statistical properties of the field variables, namely the probability and means, are of particular importance in the theory of turbulence.

We characterize these random variables using *probability distribution functions* (pdfs). For a random variable U, its pdf ρ is defined to be the derivative of $F(V)$, the *cumulative distribution function* (cdf)

$$\rho(V) := \frac{dF(V)}{dV},$$

where $F(V)$ is defined by

$$F(V) = P(U < V).$$

Then the *mean* (or *expectation*) of U is given by

$$\langle U \rangle := \int_{-\infty}^{\infty} V \rho(V)\, dV, \tag{5.1}$$

the probability-weighted average of all possible values of U. Henceforth, we shall always use $\langle \cdot \rangle$ to denote the probability averaged mean.

With the tools of probability, we can study the statistics of the random variables. However, the probability cannot be measured exactly—it can only be estimated with some confidence level. In experiments and simulations, we commonly use the time average to approximate the probability averaged mean, for processes that are statistically stationary.

For a statistically stationary system, the time average is defined as

$$\langle U(t)\rangle_T := \frac{1}{T}\int_0^T U(t)dt,$$

where $U(t)$ is the instantaneous velocity in a realization of the turbulent flow, $\langle\cdot\rangle_T$ denotes the time averaged mean with averaging time T. Under the assumption that the system is statistically steady, we know that

$$\langle\langle U(t)\rangle_T\rangle = \langle U(t)\rangle,$$

that is, the expected value of the time averaged mean $\langle U(t)\rangle_T$ agrees with the probability averaged mean $\langle U(t)\rangle$. Denoting $dU(t) = U(t) - \langle U(t)\rangle$ as the fluctuation in $U(t)$, we consider the variance of $\langle U(t)\rangle_T$,

$$var(\langle U(t)\rangle_T) = \langle(\frac{1}{T}\int_0^T dU(t)dt)^2\rangle = \frac{1}{T}\int_0^T\int_0^T \langle dU(t_1)dU(t_2)\rangle\, dt_1dt_2,$$

and thus,

$$var(\langle U(t)\rangle_T) = \frac{\sigma^2}{T}\int_{-T}^T (1-\frac{|\tau|}{T})\rho(\tau)\,d\tau \leq \frac{\Theta}{T}\sigma^2,$$

where σ is the standard deviation of $U(t)$, $\rho(\tau)$ is the correlation coefficient at time separation τ, and $\Theta = \int_{-\infty}^{\infty}|\rho(\tau)|\,d\tau < \infty$ is the correlation time. For a detailed proof, see for instance [50]. We conclude:

$$\sigma(\langle U(t)\rangle_T) = \sqrt{var(\langle U(t)\rangle_T)} \leq (\frac{\Theta}{T})^{1/2}\sigma \to 0 \qquad \text{as} \quad T\to\infty,$$

that is, if T tends to infinity, the time average $\langle U(t)\rangle_T$ converges to the mean as its standard deviation decreases to 0 at the order of $T^{-1/2}$. Thus, in the analysis of turbulent flows, we will approximate the quantities of interest (the probability averaged means) with the time average.

5.2 Formulations of shape derivatives for turbulent flows

As discussed in Chapter 3, we study the total drag exerted on the fluid to explore the influence of roughness on the flow. For turbulent flows, the mean value of the drag is given by

$$\langle \int_{\partial\Omega} \mathbf{u}\cdot(\boldsymbol{\tau}\mathbf{n})\, d\mathbf{s} + \int_{\Omega} \mathbf{u}\cdot\mathbf{f}\, d\mathbf{x}\rangle = \int_{\Omega} \Big\langle \frac{\nu}{2}(\nabla\mathbf{u}+\nabla\mathbf{u}^T) : (\nabla\mathbf{u}+\nabla\mathbf{u}^T)\Big\rangle\, d\mathbf{x}.$$

This defines the *roughness functional* in the turbulent case

$$\mathscr{D}(\Gamma_b) = \frac{\nu}{4}\int_{\Omega} \big\langle(\nabla\mathbf{u}+\nabla\mathbf{u}^T) : (\nabla\mathbf{u}+\nabla\mathbf{u}^T)\big\rangle\, d\mathbf{x}, \tag{5.2}$$

where $\mathbf{u}$ is the solution of the transient Navier-Stokes equations:

$$\begin{aligned}
\frac{\partial\mathbf{u}}{\partial t} - \nabla\cdot\big(-Ip+\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\big) + \mathbf{u}\cdot\nabla\mathbf{u} - \mathbf{f} &= 0 && \text{in } \Omega\times(0,T),\\
\nabla\cdot\mathbf{u} &= 0 && \text{in } \Omega\times(0,T),\\
\mathbf{u} &= \mathbf{u}_I && \text{on } \Omega\times\{t=0\},\\
\mathbf{u} &= \mathbf{u}_0 && \text{on } \Gamma_b\times(0,T),\\
\mathbf{u} &= \mathbf{0} && \text{on } \Gamma_t\times(0,T),
\end{aligned} \tag{5.3}$$

for any time T.

To compute the shape derivatives of (5.2), we need take variations of $\mathscr{D}(\Gamma_b)$ with respect to all variables. However, we cannot take variations inside of the mean operator directly. Thus, we use the time average as an approximation, i.e.,

$$\mathscr{D}_T(\Gamma_b) = \frac{\nu}{4T}\int_0^T \int_\Omega (\nabla \mathbf{u} + \nabla \mathbf{u}^T) : (\nabla \mathbf{u} + \nabla \mathbf{u}^T)\, d\mathbf{x}dt. \tag{5.4}$$

Since $\mathscr{D}_T(\Gamma_b)$ converges to $\mathscr{D}(\Gamma_b)$ as T tends to infinity, we want to study (5.4) to obtain the shape derivatives of (5.2). In the following, we present the formulations of the shape gradient and shape Hessian as well as the time-dependent state, adjoint and incremental equations for (5.4).

Considering the additional time axis, we define the Sobolev functional spaces for all variables:

$$\begin{aligned}
\tilde{\mathbf{H}}^1 &= \{\mathbf{u} \in L^2(0,T;\mathbf{H}^1(\Omega)) : \mathbf{u} \text{ is periodic on } \Gamma_p\},\\
\tilde{\mathbf{H}}^1_{\mathbf{u}_0} &= \{\mathbf{u} \in L^2(0,T;\tilde{\mathbf{H}}^1(\Omega)) : \mathbf{u} = \mathbf{u}_0 \text{ on } \Gamma_t\},\\
\tilde{\mathbf{H}}^1_0 &= \{\mathbf{u} \in L^2(0,T;\tilde{\mathbf{H}}^1(\Omega)) : \mathbf{u} = \mathbf{0} \text{ on } \Gamma_t\}.
\end{aligned}$$

Following the previous notations for the Lagrange multiplier functions $\boldsymbol{\lambda}, \alpha$ and $\boldsymbol{\beta}$ (normalized with T), we now define the Lagrange functional as

$$\begin{aligned}
\mathscr{L}_T(\Gamma_b, \mathbf{u}, p, \boldsymbol{\lambda}, \alpha, \boldsymbol{\beta}) &= \frac{\nu}{4T}\int_0^T \int_\Omega (\nabla \mathbf{u} + \nabla \mathbf{u}^T) : (\nabla \mathbf{u} + \nabla \mathbf{u}^T)\, d\mathbf{x}dt\\
&+ \frac{1}{T}\int_0^T \int_\Omega \Big(-\nabla\cdot\boldsymbol{\lambda} p + \frac{\nu}{2}(\nabla\boldsymbol{\lambda} + \nabla\boldsymbol{\lambda}^T) : (\nabla \mathbf{u} + \nabla \mathbf{u}^T) + (\mathbf{u}\cdot\nabla\mathbf{u})\boldsymbol{\lambda}\Big)\, d\mathbf{x}dt\\
&- \frac{1}{T}\int_0^T \int_{\Gamma_b} \boldsymbol{\lambda}\big(-Ip + \nu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)\big)\mathbf{n}\, d\mathbf{s}dt + \frac{1}{T}\int_0^T \int_{\Gamma_b} \boldsymbol{\beta}\cdot\mathbf{u}\, d\mathbf{s}dt\\
&- \frac{1}{T}\int_0^T \int_\Omega \alpha\nabla\cdot\mathbf{u}\, d\mathbf{x}dt - \frac{1}{T}\int_0^T \int_\Omega \boldsymbol{\lambda}\cdot\mathbf{f}\, d\mathbf{x}dt\\
&+ \frac{1}{T}\int_0^T \int_\Omega \frac{\partial \mathbf{u}}{\partial t}\cdot\boldsymbol{\lambda}\, d\mathbf{x}dt + \frac{1}{T}\int_\Omega (\mathbf{u} - \mathbf{u}_I)\cdot\boldsymbol{\lambda}\Big|_{t=0} d\mathbf{x}.
\end{aligned} \tag{5.5}$$

where $\mathbf{u} \in \tilde{\mathbf{H}}^1_{\mathbf{u}_0}(\Omega), p \in L^2(0,T;L^2(\Omega))$, $\boldsymbol{\lambda} \in \tilde{\mathbf{H}}^1_0(\Omega)$, $\alpha \in L^2(0,T;L^2(\Omega))$ and $\boldsymbol{\beta} \in L^2(0,T;\mathbf{H}^{-1/2}(\Gamma_b))$.

Now we derive the first-order optimality conditions of (5.5). The state equations are the transient Navier-Stokes equations given in (5.3). The adjoint equations, which are time-dependent, are given as

$$\begin{aligned} -\frac{\partial \boldsymbol{\lambda}}{\partial t} - \nabla \cdot \Big(-\mathrm{I}\,\alpha + \nu(\nabla \mathbf{u} + \nabla \mathbf{u}^T) + \nu(\nabla \boldsymbol{\lambda} + \nabla \boldsymbol{\lambda}^T)\Big) & \\ + \nabla \mathbf{u}^T \boldsymbol{\lambda} - \nabla \boldsymbol{\lambda} \mathbf{u} = \mathbf{0} & \quad \text{in } \Omega \times (0,T), \\ \nabla \cdot \boldsymbol{\lambda} = 0 & \quad \text{in } \Omega \times (0,T), \end{aligned} \tag{5.6}$$

with the terminal condition $\boldsymbol{\lambda} = 0$ at $t = T$, and $\boldsymbol{\lambda} = 0$ on $\Gamma_t \cup \Gamma_b$ and periodic on Γ_p. While the state equations evolve forward in time, the adjoint equations for $\boldsymbol{\lambda}$ evolve backward in time since the adjoint variables are known at $t = T$. The shape gradient in shape direction $\mathbf{V}$ is given by

$$\begin{aligned} & d\mathscr{D}_T(\Gamma_b;\mathbf{V}) \\ = & \frac{1}{T}\int_0^T \int_{\Gamma_b} \Big(-\frac{\nu}{4}(\nabla \mathbf{u} + \nabla \mathbf{u}^T) - \frac{\nu}{2}(\nabla \boldsymbol{\lambda} + \nabla \boldsymbol{\lambda}^T)\Big) : (\nabla \mathbf{u} + \nabla \mathbf{u}^T)\mathbf{V} \cdot \mathbf{n}\, d\mathbf{s}\, dt. \end{aligned} \tag{5.7}$$

To compute the shape Hessian, we again derive the incremental state and incremental adjoint equations by taking another variation of the state and adjoint equations with respect to all the variables. The time-dependent incremental state equations are

$$\begin{aligned} \frac{\partial \hat{\mathbf{u}}}{\partial t} - \nabla \cdot \big(-\mathrm{I}\,\hat{p} + \nu(\nabla \hat{\mathbf{u}} + \nabla \hat{\mathbf{u}}^T)\big) + \mathbf{u} \cdot \nabla \hat{\mathbf{u}} + \hat{\mathbf{u}} \cdot \nabla \mathbf{u} = \mathbf{0} & \quad \text{in } \Omega \times (0,T), \\ \nabla \cdot \hat{\mathbf{u}} = 0 & \quad \text{in } \Omega \times (0,T), \qquad (5.8) \\ \hat{\mathbf{u}} = 0 & \quad \text{on } \Omega \times \{t = 0\}, \end{aligned}$$

$$\hat{\mathbf{u}} + (\nabla\mathbf{u} + \nabla\mathbf{u}^T)\mathbf{n}\mathbf{W}\cdot\mathbf{n} = \mathbf{0} \qquad \text{on } \Gamma_b \times (0,T),$$

and the incremental adjoint equations

$$\begin{aligned}
-\frac{\partial\hat{\boldsymbol{\lambda}}}{\partial t} - \nabla\cdot\Big(-\mathrm{I}\,\hat{\alpha} + \nu(\nabla\hat{\mathbf{u}} + \nabla\hat{\mathbf{u}}^T) + \nu(\nabla\hat{\boldsymbol{\lambda}} + \nabla\hat{\boldsymbol{\lambda}}^T)\Big) &= \\
-\nabla\hat{\mathbf{u}}^T\boldsymbol{\lambda} - \nabla\mathbf{u}^T\hat{\boldsymbol{\lambda}} + \nabla\boldsymbol{\lambda}\hat{\mathbf{u}} + \nabla\hat{\boldsymbol{\lambda}}\mathbf{u} &\qquad \text{in } \Omega\times(0,T), \\
\nabla\cdot\hat{\boldsymbol{\lambda}} = 0 &\quad \text{in } \Omega\times(0,T), \\
\hat{\boldsymbol{\lambda}} = 0 &\quad \text{on } \Omega\times\{t=T\}, \\
\hat{\boldsymbol{\lambda}} + (\nabla\boldsymbol{\lambda} + \nabla\boldsymbol{\lambda}^T)\mathbf{n}(\mathbf{W}\cdot\mathbf{n}) = 0 &\quad \text{on } \Gamma_b\times(0,T),
\end{aligned} \tag{5.9}$$

where $\mathbf{W}$ is the second variation of the shape Γ_b, $\hat{\mathbf{u}} = 0$, $\hat{\boldsymbol{\lambda}} = 0$ on Γ_b, and they are periodic in Γ_p. Thus the shape Hessian evaluated in the directions $(\mathbf{V},\mathbf{W})$ gives

$$\begin{aligned}
d^2\mathscr{D}_T(\Gamma_b;\mathbf{V},\mathbf{W}) = \frac{1}{T}\int_0^T\int_{\Gamma_b}\Big(&-\frac{\nu}{2}(\nabla\mathbf{u}+\nabla\mathbf{u}^T) : (\nabla\hat{\mathbf{u}}+\nabla\hat{\mathbf{u}}^T) - \frac{\nu}{2}(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T) : \\
&(\nabla\hat{\mathbf{u}}+\nabla\hat{\mathbf{u}}^T) - \frac{\nu}{2}(\nabla\hat{\boldsymbol{\lambda}}+\nabla\hat{\boldsymbol{\lambda}}^T) : (\nabla\mathbf{u}+\nabla\mathbf{u}^T)\Big)\mathbf{V}\cdot\mathbf{n}\,d\mathbf{s}\,dt \\
&+\frac{1}{T}\int_0^T\int_{\Gamma_b}\nabla\cdot\Big(-\frac{\nu}{4}(\nabla\mathbf{u}+\nabla\mathbf{u}^T) : (\nabla\mathbf{u}+\nabla\mathbf{u}^T)\mathbf{V}\Big)(\mathbf{W}\cdot\mathbf{n})\,d\mathbf{s}\,dt \\
&+\frac{1}{T}\int_0^T\int_{\Gamma_b}\nabla\cdot\Big(-\frac{\nu}{2}(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T) : (\nabla\mathbf{u}+\nabla\mathbf{u}^T)\mathbf{V}\Big)(\mathbf{W}\cdot\mathbf{n})\,d\mathbf{s}\,dt.
\end{aligned} \tag{5.10}$$

With the formulations (5.7) and (5.10), now we want to solve the state, adjoint and incremental equations to compute the shape gradient and shape Hessian.

5.3 Evaluation of shape derivatives

For high Reynolds number flow, analyzing the effect of roughness on turbulence, computing the shape gradient and shape Hessian is computationally costly.

Additional algorithmic challenges originate from the space-time coupling of the transient Navier-Stokes equations (5.3) and their adjoint (5.6).

The solution of (5.3) for the turbulent velocity in a channel flow is well established [17, 31, 40, 53]. In the standard numerical approach, the formulation of the Navier-Stokes equations used by Kim *et al* [40] is solved using a Fourier Galerkin formulation (dealiased spectral method) for the directions parallel to the wall and a Chebychev-τ method for the wall-normal directions. In the current study, the treatment of the adjoint problem is facilitated by the use of a Legendre Galerkin representation of the wall-normal direction. For the time discretization, a low-storage, second-order implicit-explicit Runge-Kutta scheme is used for the forward systems and a consistent adjoint scheme is applied to the backward-in-time adjoint equations. A direct numerical simulation (DNS) code using the formulation in [40] which supports Galerkin Legendre and B-spline formulations has been developed in [48]. For the present work it has been extended by the adjoint and incremental solvers. For a detailed description of the numerical algorithms used, we refer to Appendix A for the forward solver, Appendix B for the backward solver and Chapter 6 for the IMEX schemes used for the time discretization.

The adjoint equations (5.6) and (5.9) must be solved backwards in time due to the terminal condition. Since the adjoint equations are linearized around the state solutions, integrating the adjoint equation requires knowledge of the complete time history of the state solutions $\mathbf{u}$ and $\hat{\mathbf{u}}$. Since a meaningful simulation will need more than 10^4 time steps, keeping all the time steps of the state solutions in memory is impossible, which is a known problem when computing adjoints of time-dependent systems. As a remedy, we use a checkpointing procedure (see for instance [24]), in which at selected time steps snapshots of $\mathbf{u}$ and $\hat{\mathbf{u}}$ are stored on disk. Then, when needed in the backwards integration of the adjoint equations, time step values of $\mathbf{u}$ and $\hat{\mathbf{u}}$ are recomputed based on the stored snapshots. This procedure increases the cost of a shape Hessian computation from approximately four times the cost for a Navier-Stokes solve to roughly six times a Navier-Stokes solve. However, these approaches have been shown to be efficient in practice since they balance memory and computational resources, and also avoid massive I/O to and from disk.

To compute the roughness gradient and Hessian, we first solve the linear incremental state equation (5.8) along with the Navier-Stokes (state) equations (5.3), which are conveniently solved together. The adjoint (5.6) and incremental adjoint (5.9) equations are then solved together and the gradient and Hessian are evaluated using (5.7) and (5.10). As in the stationary case (Chapter 4), it can be shown that for a flat boundary $\Gamma_b = 0$ the eigenfunctions of the shape Hessian are the Fourier modes. This implies that the shape Hessian is diagonal in the Fourier basis and, hence, one incremental state and adjoint solution is sufficient to find all its eigenvalues: We simply choose W as a sum of eigenfunctions and then use that $\gamma(k) = d^2_{qq}\mathscr{D}(W, V_k)$, where V_k is the k-th Fourier function. The complete determination of the flat-boundary Hessian can thus be accomplished with just two state and two adjoint solutions.

Now with the solutions to the four systems obtained from the DNS, in principle, we are ready to compute the shape gradient and shape Hessian for the turbulent case. Solved from the forward solver, the velocity profiles for state equations

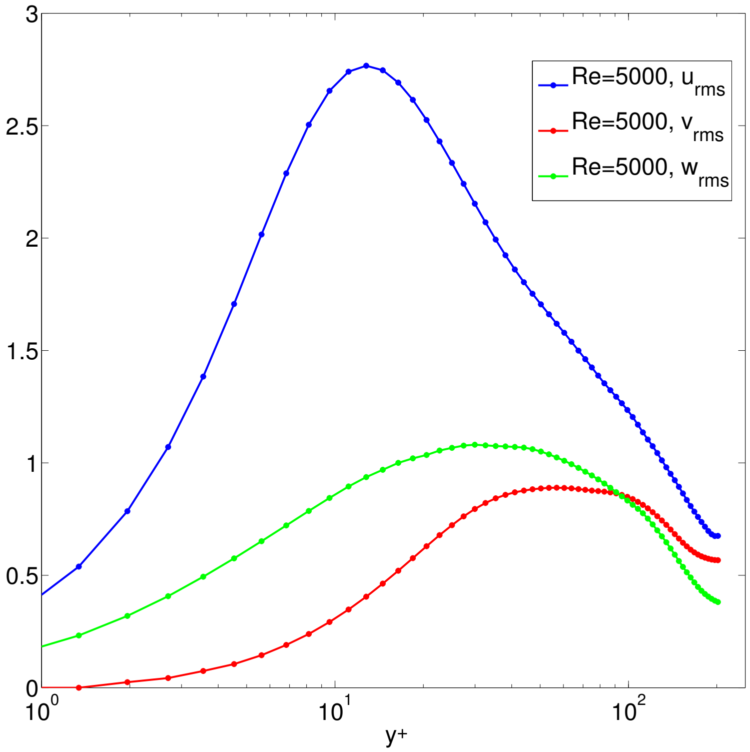

Figure 5.1: Root-mean-square velocity fluctuations normalized by the wall shear velocity for Reynolds number $Re_h = 5000$.

(turbulent flows) at $Re_h = 3000$ or $Re_h = 5000$ (where h is the height of the channel) are shown in Fig. 5.1 for the turbulence intensities normalized by the friction velocity, Fig. 5.2 for the mean velocity profile versus the law of the wall, and Fig. 5.3 for Reynolds shear stress. These behaviors match well with the the velocity profiles from DNS of channel flow by Kim *et all.* in [40].

Solved using the adjoint solver, at $Re = 3000$, with terminal time $T = 1000$, the solutions to the adjoint equations are given in Fig. 5.4. It shows that the adjoint velocities grow exponentially backward in time. At the two different locations, near

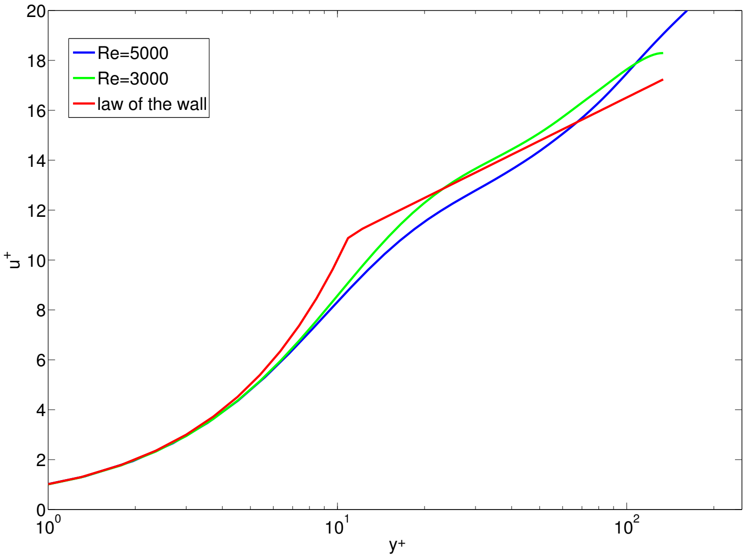

Figure 5.2: Mean velocity profile u^+ compared with law of the wall for turbulent channel flows with $Re_h = 3000$ and $Re_h = 5000$.

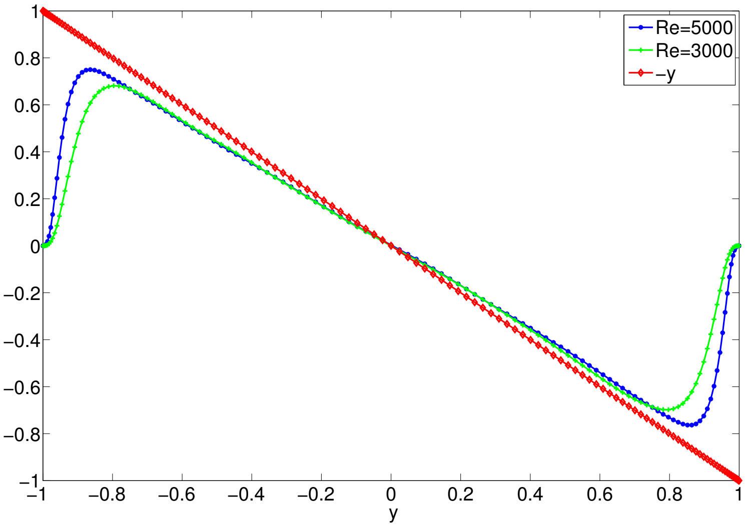

Figure 5.3: Reynolds shear stress normalized by the wall shear velocity in global coordinates.

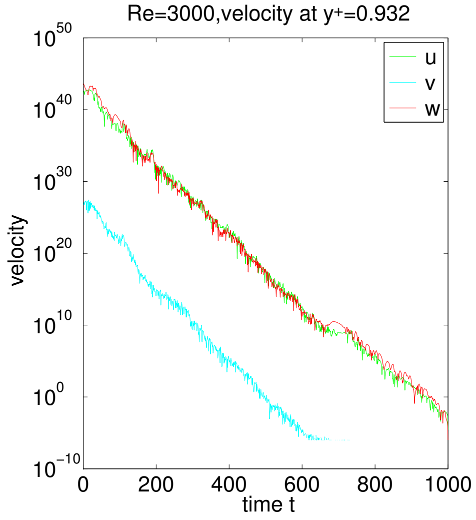

Figure 5.4: The backward growth of adjoint velocities at $y^+ = 0.932$ and $y^+ = 60.4$ at $Re_h = 3000$.

the wall ($y^+ = 0.932$), away from the wall ($y^+ \approx 60$), the magnitude of the velocities could be of order 10^{40} after a developing time $T \approx 1000$. As a direct result,

$$\frac{1}{T}\int_0^T \int_{\Gamma_b} \left(\frac{\nu}{2}(\nabla \boldsymbol{\lambda} + \nabla \boldsymbol{\lambda}^T)\right) : (\nabla \mathbf{u} + \nabla \mathbf{u}^T)\mathbf{V} \cdot \mathbf{n}\, d\mathbf{s}\, dt \to \infty \qquad \text{as } T \to \infty,$$

that is, $d\mathscr{D}_T(\Gamma_b; \mathbf{V})$ is not converging as $T \to \infty$ and we cannot use the time average to approximate the mean values. So the direct application of the adjoint method to compute the shape derivatives for turbulent flows fails.

We notice the same behaviors for the incremental state and incremental adjoint solutions as well. The reason for the rapid growth is the chaotic nature of turbulent flows. As a chaotic system, turbulent flows are very sensitive to small changes in the initial conditions or boundary conditions, i.e., a small perturbation will lead to a large difference in the instantaneous velocity profile. Thus, the adjoint and incremental variables, which represent this sensitivity, grow exponentially in time. This requires that the integration time can not be too long to have a finite and meaningful time average, which contradicts the need for a long-time average to better approximate the mean values.

To better understand this issue, we will employ the Lorenz system as a simplified model and perform sensitivity analysis on this model problem.

Chapter 6

Implicit-explicit Runge-Kutta scheme

In this chapter, we study the implicit-explicit (IMEX) Runge-Kutta time discretization used in the forward and backward numerical simulations, and derive a special IMEX scheme that is both gradient consistent (i.e., the discretized time-continuous gradient is equivalent to the gradient of the discretized objective function) as well as adjoint consistent (i.e., the accuracy of the implied adjoint discretization is of the same order as for the forward scheme). *Runge-Kutta* (RK) methods are an important family of numerical methods used to solve systems of ordinary differential equations (ODEs). For an initial value problem

$$\begin{aligned} y' &= f(t,y), \\ y(t_0) &= y_0, \end{aligned} \tag{6.1}$$

an s-stage RK method stepping from t_n to t_{n+1} is as follows:

$$y_{n+1} = y_n + h \sum_{j=1}^{s} b_i k_i, \tag{6.2}$$

where

$$k_i = f(t_n + c_i h, y_n + h \sum_{j=1}^{s} a_{ij} k_j).$$

Here, $h = t_{n+1} - t_n$ is the time step size, $k_1, \cdots, k_s$ are the stages and $b_1, \cdots, b_s, a_{ij}(i, j = 1, \cdots s), c_i = \sum_j a_{ij}$ are the coefficients characterizing the RK method, which can be summarized in the *Butcher tableau* as

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^T \end{array} = \begin{array}{c|ccc} c_1 & a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & & \vdots \\ c_s & a_{s1} & \cdots & a_{ss} \\ \hline & b_1 & \cdots & b_s \end{array}.$$

The RK scheme (6.2) is an *explicit scheme* if the matrix $A = (a_{ij})$ is a lower-triangular matrix (i.e., $a_{ij} = 0$, for $i \geq j$); otherwise, it is called an *implicit scheme*. For a complete discussion of Runge-Kutta methods, we refer to [14].

An essential part of our sensitivity analysis for the turbulent flow case is to numerically solve the Navier-Stokes equations, as well as their linearized forms occurring in first- and second-order derivative computations. As described in Appendix A and B, we first discretize in space to convert the time-dependent partial differential equations (for instance, (A.1)) to a system of ordinary differential equations ((A.18)-(A.21)). Then, we discretize in time using Runge-Kutta methods. Here, we focus on the time discretization scheme used in the numerical simulations.

Consider a system of ODEs of the form

$$\begin{aligned} \frac{\partial u}{\partial t} &= L(u) + N(u), \\ u(t_0) &= u_0, \end{aligned} \tag{6.3}$$

which arises from spatial discretization of PDEs of convective-diffusive nature. Here, u is a state variable, N is possibly nonlinear in u which corresponds to the convection (hyperbolic) term, and L corresponds to the diffusion (parabolic) term. Since N is often nonlinear and the Jacobian of N is nonsymmetric and indefinite, integrating the ODE (6.3) implicitly requires the inversion of this Jacobian at each time step, which would be very expensive. Thus, we prefer to treat N explicitly, particularly since the CFL-limited time step is of the order of that required for accuracy. On the other hand, $L(u)$ is a stiff and linear term which has to be integrated implicitly to avoid excessively small time steps. Thus, we choose to discretize L implicitly and N explicitly, yielding an implicit-explicit Runge-Kutta scheme.

In the following, we first summarize the order conditions of the IMEX scheme. Then the time discretization scheme for the adjoint equation is derived. This derivation is based on the discrete adjoint of the forward time discretization. The adjoint time discretization scheme is studied and its application for DNS simulations is discussed.

6.1 IMEX method

For the ODE given in (6.3), or more generally

$$\begin{aligned} \frac{\partial u}{\partial t} &= L(t,u) + N(t,u), \\ u(t_0) &= u_0, \end{aligned} \tag{6.4}$$

a general form of an IMEX scheme with s stages for the time interval $[t_n, t_{n+1}]$ is given by

$$u_{n+1} = u_n + h \sum_{j=1}^{s} b_j L(t_n + hc_j, Y_j) + h \sum_{j=1}^{s} \hat{b}_j N(t_n + h\hat{c}_j, Y_j), \tag{6.5}$$

with

$$Y_i = u_n + h \sum_{j=1}^{s} a_{ij} L(t_n + hc_j, Y_j) + h \sum_{j=1}^{s} \hat{a}_{ij} N(t_n + h\hat{c}_j, Y_j), \qquad \text{for } i = 1, \cdots, s.$$

Here, $(\hat{a}_{ij})$ is a lower-triangular matrix with $\hat{a}_{ij} = 0$ for $j \geq i$, while $a_{ij} \neq 0$ for $j = i$. Thus, the scheme is explicit in N and implicit in L. In general, in order to guarantee simplicity and efficiency in solving the algebraic equations corresponding to the implicit part of the discretization at each step, we consider diagonally implicit Runge-Kutta (DIRK) methods for L (see [11]), i.e., $a_{ij} = 0$ for $j > i$. $c_i = \sum_{j=1}^{i} a_{ij}$ and $\hat{c}_i = \sum_{j=1}^{i-1} \hat{a}_{ij}$ are the substeps for L and N. The coefficients $a_{ij}, \hat{a}_{ij}, b_j, \hat{b}_j$ are chosen such that the scheme satisfies the desired accuracy. We require L and N to have the same substeps in (6.5) to be time consistent, i.e., $c_i = \hat{c}_i$, for all $i = 1, \cdots, s$. Characterized by the coefficients vectors $\mathbf{A}, \hat{\mathbf{A}}, \mathbf{c}, \hat{\mathbf{c}}, \mathbf{b}, \hat{\mathbf{b}}$, the IMEX method (6.5) can be represented by a double *tableaus* in the usual Butcher notation:

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^T \end{array} \qquad \begin{array}{c|c} \hat{\mathbf{c}} & \hat{\mathbf{A}} \\ \hline & \hat{\mathbf{b}}^T \end{array}.$$

Notice that we can recast the system (6.4) as:

$$\begin{aligned} \frac{\partial v}{\partial t} &= \hat{L}(v,w,t) := L(v+w,t), \\ \frac{\partial w}{\partial t} &= \hat{N}(v,w,t) := N(v+w,t), \end{aligned} \tag{6.6}$$

with $u = v + w$. The IMEX scheme (6.5) applied to this problem is then a partitioned Runge-Kutta scheme. Thus, IMEX schemes can be viewed as a particular class of partitioned Runge-Kutta methods and the general analysis of partitioned methods is applicable; see [27] for the analysis of order conditions for these schemes.

6.1.1 Order conditions

The general technique to derive order conditions for Runge-Kutta schemes is to compare the Taylor expansions of the exact and numerical solutions. In particular, a scheme is said to be of order p if the numerical solution at $t = t_0 + h$ obtained with the given initial condition at time t_0 agrees with the real solution up to order h^{p+1}. Applying (6.5) to the ODE system (6.4), we can conclude the following order conditions:

- First order:

$$\sum_{i=1}^{s} b_i = 1, \qquad \sum_{i=1}^{s} \hat{b}_i = 1; \tag{6.7}$$

- Second order:

$$\sum_{i=1}^{s} b_i c_j = \frac{1}{2}, \qquad \sum_{i=1}^{s} \hat{b}_i \hat{c}_i = \frac{1}{2}; \tag{6.8}$$

- Third order:

$$\sum_{i,j} b_i a_{i,j} c_j = \frac{1}{6}, \quad \sum_i b_i c_i^2 = \frac{1}{3}, \tag{6.9}$$

$$\sum_{i,j} \hat{b}_i \hat{a}_{i,j} \hat{c}_j = \frac{1}{6}, \quad \sum_i \hat{b}_i \hat{c}_i^2 = \frac{1}{3}, \tag{6.10}$$

$$\sum_{i,j} \hat{b}_i a_{i,j} c_j = \frac{1}{6}, \quad \sum_{i,j} b_i \hat{a}_{i,j} c_j = \frac{1}{6}. \tag{6.11}$$

These conditions are the simplified results given our assumption that $c_i = \hat{c}_i$. Note that the conditions (6.7)-(6.10) are the standard order conditions for the two tableaus, each of them taken separately. Condition (6.11) is the new condition arising because of the coupling of the two schemes. See [56] for a more general discussion about the order conditions.

6.1.2 Gradient consistent RK scheme for the adjoint systems

As stated in the previous section, to compute gradients, we need to solve forward state systems as well as backward adjoint systems. Here, we study the time discretization schemes for the adjoint systems. To illustrate the idea of adjoint consistency, consider an optimal control problem governed by a state equation in the form of an ODE, in which the control variables are the initial condition u^*, and the objective function represents a cost associated with the final state u_N, i.e.,

$$\begin{aligned} &\min_{u^*} C(u_N) \\ \text{s.t.}\quad &\frac{\partial u}{\partial t} = L(u) + N(u), \quad \text{for } t \in [0, t_N], \\ &u(0) = u^*. \end{aligned} \tag{6.12}$$

The continuous-in-time adjoint equations are terminal value problems and usually have the following form:

$$\frac{\partial \lambda}{\partial t} = -(\nabla L^T + \nabla N^T)\lambda, \qquad \lambda(t_N) = -\nabla C(u_N), \tag{6.13}$$

where ∇ represents the Jacobian with respect to u, and $\nabla L^T, \nabla N^T$ are the adjoint operators for the linearization of L and N.

To arrive at discretized adjoint equations, there are two approaches: optimize (resulting in (6.13)) then discretize (OTD) or discretize (6.12) then optimize (DTO), which yields a discrete form of (6.13). The order that one performs optimization and discretization may result in different discretized adjoints and gradients. *Gradient consistency* imposes a constraint on the discretization of the adjoint equations (6.13) that requires the two approaches (OTD and DTO) to yield the same discretized adjoint equations; for a complete discussion, see [25]. Thus, for a given discretization scheme for the forward system (6.12), a gradient consistent adjoint discretization scheme of the continuous adjoint equations (6.13) can be found by the DTO procedure, that is, finding the adjoint system of the discretized optimization problem. In the following, we will derive a gradient consistent IMEX scheme of the form (6.5).

Let us leave the objective functional in the continuous form and discretize the state ODE following (6.5); the discretized optimal control problem becomes:

$$
\begin{aligned}
&\min_{u^*} C(u_N)\\
&\text{s.t.}\\
&u_{n+1} = u_n + h\sum_{j=1}^{s} b_j L(t_n + hc_j, Y_j) + h\sum_{j=1}^{s} \hat{b}_j N(t_n + h\hat{c}_j, Y_j) \qquad \text{for } n = 0, \cdots, N-1,\\
&Y_i = u_n + h\sum_{j=1}^{i} a_{ij} L(t_n + hc_j, Y_j) + h\sum_{j=1}^{i-1} \hat{a}_{ij} N(t_n + h\hat{c}_j, Y_j), \qquad i = 1, \cdots, s,\\
&u_0 = u^*.
\end{aligned}
\tag{6.14}
$$

To optimize the above discretized problem, we define the Lagrangian function

$$
\begin{aligned}
L(u^*, u, \psi) =& C(u_N) + \psi_0(u_0 - u^*)\\
&+ \sum_{n=0}^{N-1}\Bigg(\psi_{n+1}\Big(u_{n+1} - u_n - h\sum_{j=1}^{s} b_j L(t_n + hc_j, Y_j) - h\sum_{j=1}^{s} \hat{b}_j N(t_n + h\hat{c}_j, Y_j)\Big)\\
&\qquad + \sum_{i=1}^{s} \lambda_i \Big(Y_i - u_n - h\sum_{j=1}^{i} a_{ij} L(t_n + hc_j, Y_j) - h\sum_{j=1}^{i-1} \hat{a}_{ij} N(t_n + h\hat{c}_j, Y_j)\Big)\Bigg).
\end{aligned}
\tag{6.15}
$$

The adjoint equations can be derived by taking derivatives of the Lagrangian with respect to u_n and Y_k, i.e.,

$$\begin{aligned}
\frac{\partial L}{\partial u_n} &= -\psi_{n+1} + \psi_n - \sum_i \lambda_i = 0, \qquad\qquad \text{for } n = 0, \cdots, N-1;\\
\frac{\partial L}{\partial Y_k} &= \psi_{n+1} h\Big(-b_k \nabla L^T(t_n + hc_k, Y_k) - \hat{b}_k \nabla N^T(t_n + h\hat{c}_k, Y_k)\Big) + \lambda_k +\\
&\qquad \sum_i \lambda_i h\Big(-a_{ik} \nabla L^T(t_n + hc_k, Y_k) - \hat{a}_{ik} \nabla N^T(t_n + h\hat{c}_k, Y_k)\Big), \text{ for } k = 1, \cdots, s;\\
\frac{\partial L}{\partial u_N} &= \nabla C(u_N) + \psi_N.
\end{aligned} \tag{6.16}$$

This yields the discretized adjoint equations for the adjoint variable ψ_n and its stages λ_k,

$$\begin{aligned}
\psi_n &= \psi_{n+1} + \sum_i \lambda_i,\\
\lambda_k &= h\Big(b_k \psi_{n+1} + \sum_i a_{ik} \lambda_i\Big) \nabla L^T(t_n + hc_k, Y_k)\\
&\quad + h\Big(\hat{b}_k \psi_{n+1} + \sum_i \hat{a}_{ik} \lambda_i\Big) \nabla N^T(t_n + h\hat{c}_k, Y_k),
\end{aligned} \tag{6.17}$$

with terminal condition

$$\psi_N = -\nabla C(u_N).$$

This is also an implicit-explicit scheme, which is implicit in ∇L^T and explicit in ∇N^T.

For convenience in the following analysis, let us consider an IMEX scheme for the state ODE with a slightly changed structure: given u_n,

$$\begin{aligned} Y_1 &= u_n, \\ Y_i &= Y_{i-1} + h\sum_{j=1}^{i} a_{ij} L(t_n + hc_j, Y_j) + h\sum_{j=1}^{i-1} \hat{a}_{ij} N(t_n + h\hat{c}_j, Y_j), \quad i = 2, \cdots, s \\ Y_{s+1} &= Y_s + h\sum_{j=1}^{s} b_j L(t_n + hc_j, Y_j) + h\sum_{j=1}^{s} \hat{b}_j N(t_n + h\hat{c}_j, Y_j). \\ u_{n+1} &= Y_{s+1}. \end{aligned} \tag{6.18}$$

That is, each new stage Y_i is incremented from its previous stage Y_{i-1} instead of previous time step u_n. Then following the DTO procedure as above, we can derive the discrete adjoint scheme corresponding to (6.18) as

$$\begin{aligned} \psi_{s+1} &= \lambda_{n+1}, \\ \psi_k &= \psi_{k+1} + h\sum_{i \geq k} a_{ik} \nabla L^T(t_n + hc_k, Y_k)\psi_i + h\sum_{i>k} \hat{a}_{ik} \nabla N^T(t_n + h\hat{c}_k, Y_k)\psi_i \\ &\qquad + hb_k \nabla L^T(t_n + hc_k, y_k)\psi_{s+1} + h\hat{b}_k \nabla N^T(t_n + h\hat{c}_k)\psi_{s+1}, \\ \lambda_n &= \psi_1. \end{aligned} \tag{6.19}$$

Similarly, the adjoint stage ψ_k is also incremented from its previous stage ψ_{k+1}. Thus, by using the special structure of the forward scheme (6.18), we have created an adjoint IMEX scheme (6.19) of exactly the same structure, which is very convenient in the numerical simulations.

We now interpret the discrete adjoint equations (6.17) or (6.19) as a time discretization applied to the continuous adjoint equation (6.13), and study the accuracy of this scheme. Discrete adjoints are very popular in optimization and control, since they guarantee that resulting adjoint-based gradient is indeed the gradient of the discretized optimization problem. *Adjoint consistency* is a constraint on time discretization of the state ODE that specifies that the discrete adjoint has the same order of accuracy as the IMEX scheme for the state ODE. For the analysis of adjoint consistency, see for instance [26, 62]. It is shown in [62] that for any explicit or implicit forward scheme, its adjoint scheme has the same order of accuracy. However, this is in general not true for an IMEX scheme. In the following, we will derive the additional requirements for the adjoint consistency of a specific IMEX scheme used for the discretization of the Navier-Stokes equations.

6.2 IMEX scheme used in DNS

The time discretization scheme used in our DNS follows the method described in [67]. It is a low-storage IMEX scheme, explicit and third-order for the convection term, and implicit and second-order for the viscous term.

Consider the forward problem

$$\frac{\partial u}{\partial t} = L(u) + N(t, u),$$

where L is a linear operator and comprises the viscous term and the pressure term and does not depend on time t, and N corresponds to the convection term and external forces and may be time dependent. To advance from t_n to t_{n+1}, the following three substeps are taken:

$$\begin{aligned}
u'_n &= u_n + h[a_1 L(u_n) + b_1 L(u'_n) + c_1 N(t_n, u_n)], \\
u''_n &= u'_n + h[a_2 L(u'_n) + b_2 L(u''_n) + c_2 N(t_n + r_1 h, u'_n) + d_2 N(t_n, u_n)], \\
u_{n+1} &= u''_n + h[a_3 L(u''_n) + b_3 L(u_{n+1}) + c_3 N(t_n + r_2 h, u''_n) + d_3 N(t_n + r_1 h, u'_n)],
\end{aligned} \tag{6.20}$$

where $r_1 = a_1 + b_1 = c_1, r_2 = a_1 + b_1 + a_2 + b_2 = c_1 + c_2 + d_2, r_3 = a_1 + b_1 + a_2 + b_2 + a_3 + b_3 = c_1 + c_2 + d_2 + c_3 + d_3$ are the three stages sizes. We point out that (6.20) has the same structure as (6.18), and thus the adjoint scheme will have the structure of (6.19).

Using Taylor expansion, we can conclude the following order conditions for the above scheme.

- First order:

$$\begin{aligned} a_1 + a_2 + a_3 + b_1 + b_2 + b_3 &= 1, \\ c_1 + c_2 + c_3 + d_2 + d_3 &= 1; \end{aligned} \tag{6.21}$$

- Second order:

$$\begin{aligned} (a_2 + b_1)(a_1 + b_1) + (a_3 + b_2)(a_1 + b_1 + a_2 + b_2) + b_3 &= \frac{1}{2}, \\ (c_2 + d_3)c_1 + c_3(c_1 + c_2 + d_2) &= \frac{1}{2}; \end{aligned} \tag{6.22}$$

- Third order:

$$\begin{aligned} c_1 c_2 c_3 &= \frac{1}{6}, \\ c_1^2(c_2 + d_3) + c_3(c_1 + c_2 + d_2)^2 &= \frac{1}{3}, \\ b_3[(c_3(c_1 + c_2 + d_2) + c_1 d_3] + c_1 c_2(c_3 + d_3) + c_1 c_2 b_2 &= \frac{1}{6}, \\ c_3[c_1 b_1 + (c_2 + d_2)(c_1 + b_2)] + c_1 b_1(c_2 + d_3) &= \frac{1}{6}, \\ [(c_1 + b_3 + b_2)(d_2 + c_2) + (b_3 + b_1)c_1 + b_3^2]d_3 & \\ +[(c_1 + b_3 + b_2)c_3 + (b_2 + b_1)c_1 + b_2^2]d_2 & \\ +[(c1 + b_3 + b_2)c_2 + (b_3 + b_1)c_1 + b_3^2]c_3 & \\ +[(b_2 + b_1)c_1 + b_2^2]c_2 + b_1^2 c_1 &= \frac{1}{6}. \end{aligned} \tag{6.23}$$

Let us derive the adjoint scheme following the DTO procedure for (6.20). Consider an optimal control problem,

$$\begin{aligned} &\min\nolimits_{u^*} Q = \textstyle\int_0^T j(u,u^*)\,dt \\ \text{s. t.}& \\ &\tfrac{\partial u}{\partial t} = L(u) + N(t,u), \\ &u(0) = u^*. \end{aligned} \tag{6.24}$$

its adjoint problem is:

$$\begin{aligned} &-\frac{\partial \lambda}{\partial t} = \nabla L^T \lambda + \nabla N(t,u)^T \lambda + \nabla j^T, \\ &\lambda(N) = 0; \end{aligned} \tag{6.25}$$

To discretize the objective functional in a way that will lead to gradient consistency, we introduce a new variable $J(t)$, where

$$J(t) = \int_0^t j(u(s),u^*)\,ds, \quad \text{then} \quad \frac{\partial J}{\partial t} = j(u(t),u^*), \quad J(0) = 0,$$

and then (6.24) is equivalent to

$$\begin{aligned} &\min_{u^*} J(T) \\ \text{s. t.}& \\ &\frac{\partial u}{\partial t} = L(u) + N(t,u), \\ &\frac{\partial J}{\partial t} = j(u), \\ &u(0) = u^*, \\ &J(0) = 0. \end{aligned} \tag{6.26}$$

Now applying DTO to (6.26) gives us the discretized adjoint system. As described in (6.20), the linear operator L and the nonlinear operator N are treated implicitly and explicitly, respectively. The operator j in the objective functional is the dissipation of energy, thus, it is also discretized implicitly as the diffusion term. The discretized optimal control problem is:

$$\begin{aligned}
&\min_{u^*} J(T)\\
&\text{s. t.} \quad \text{for all } n\\
u'_n &= u_n + h\Big(L(a_1u_n + b_1u'_n) + c_1N(t_n,u_n)\Big),\\
u''_n &= u' + h\Big(L(a_2u'_n + b_2u''_n) + c_2N(t_n+hr_1,u') + d_2N(t_n,u_n)\Big),\\
u_{n+1} &= u'' + h\Big(L(a_3u'' + b_3u_{n+1} + c_3N(t_n+hr_2,u'') + d_3N(t_n+hr_1,u')\Big),\\
J'_n &= J_n + h\Big(a_1j(u_n) + b_1j(u'_n)\Big),\\
J''_n &= J'_n + h\Big(a_2j(u'_n) + b_2j(u''_n)\Big),\\
J_{n+1} &= J''_n + h\Big(a_3j(u''_n) + b_3j(u_{n+1})\Big),\\
u_0 &= u^*,\\
J_0 &= 0.
\end{aligned} \tag{6.27}$$

The first-order optimality conditions of (6.27) give the discrete adjoint systems:

$$\begin{aligned}
\lambda''_n &= \lambda_n + h\Big(a_3\nabla L^T(u''_n)\lambda_n + b_2\nabla L^T(u''_n)\lambda''_n + a_3\nabla j^T(u''_n) + b_2\nabla j^T(u''_n)\\
&\qquad\qquad + c_3\nabla N^T(t_n+hr_2,u''_n)\lambda_n\Big),\\
\lambda'_n &= \lambda''_n + h\Big(a_2\nabla L^T(u'_n)\lambda''_n + b_1\nabla L^T(u'_n)\lambda'_n + a_2\nabla j^T(u'_n) + b_1\nabla j^T(u'_n)\\
&\qquad\qquad + c_2\nabla N^T(t_n+hr_1,u'_n)\lambda''_n + d_3\nabla N^T(t_n+hr_1,u'_n)\lambda_n\Big),\\
\lambda_{n-1} &= \lambda'_n + h\Big(a_1\nabla L^T(u_n)\lambda'_n + b_3\nabla L^T(u_n)\lambda_{n-1}a_1\nabla j^T(u_n) + b_3\nabla j^T(u_n)\\
&\qquad\qquad + c_1\nabla N^T(u_n)\lambda'_n + d_2\nabla N^T(u_n)\lambda''_n\Big),
\end{aligned} \tag{6.28}$$

with the terminal condition $\lambda_N = 0$, and here ∇L^T, ∇N^T, and ∇j^T are the corresponding adjoint operators. We can also conclude the quadrature rule for the integration of Q as:

$$Q = \int_0^T j(\mathbf{u})\,dt = \sum_n h\Big(a_1 j(\mathbf{u}_n) + (a_2+b_1) j(\mathbf{u}_n') + (a_3+b_2) j(\mathbf{u}_n'') + b_3 j(\mathbf{u}_{n+1})\Big).$$

By Taylor expansion again, we can conclude the order conditions for (6.28):

- First order:

$$\begin{aligned} a_1 + a_2 + a_3 + b_1 + b_2 + b_3 &= 1, \\ c_1 + c_2 + c_3 + d_2 + d_3 &= 1; \end{aligned}$$

- Second order:

$$\begin{aligned} (a_2+b_1)(a_1+b_1) + (a_3+b_2)(a_1+b_1+a_2+b_2) + b_3 &= \frac{1}{2}, \\ (c_2+d_3)c_1 + c_3(c_1+c_2+d_2) &= \frac{1}{2}, \\ (a_2+b_2)c_3 + (a_1+b_1)(c_3+c_2+d_3) + b_3 &= \frac{1}{2}, \\ (c_2+d_2)(a_3+b_2) + c_1(a_3+b_2+a_2+b_1) &= \frac{1}{2}. \end{aligned}$$

Notice that the stage sizes for ∇L^T and ∇N^T are

$$\begin{pmatrix} a_3+b_2 \\ a_3+b_2+a_2+b_1 \\ a_3+b_2+a_2+b_1+a_1+b_3 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} c_3 \\ c_3+c_2+d_3 \\ c_3+c_2+d_3+c_1+d_2 \end{pmatrix}$$

respectively. Although we assume that the stages sizes for L and N are the same at each substep to be time consistent in the forward IMEX scheme, generally this assumption is not satisfied for the discrete adjoint scheme given in (6.28). The additional assumption that the adjoint scheme is also time step consistent, gives us an important set of sufficient second-order conditions:

$$\begin{aligned}(a_2+b_1)(a_1+b_1)+(a_3+b_2)(a_1+b_1+a_2+b_2)+b_3 &= \frac{1}{2},\\ (c_2+d_3)c_1+c_3(c_1+c_2+d_2) &= \frac{1}{2},\\ a_3+b_2-c_3 &= 0,\\ a_2+b_1-c_2-d_3 &= 0,\\ a_1+b_3-c_1-d_2 &= 0.\end{aligned}$$

Adjoint consistency requires that we have the same accuracy for the state and adjoint ODEs, i.e., the order conditions need to be satisfied at the same time. Thus, to have the second-order accuracy, sufficient conditions for the coefficients a_i, b_i, c_i, d_i are:

$$\begin{aligned}a_1+a_2+a_3+b_1+b_2+b_3 &= 1,\\ c_1+c_2+c_3+d_2+d_3 &= 1,\\ (a_2+b_1)(a_1+b_1)+(a_3+b_2)(a_1+b_1+a_2+b_2)+b_3 &= \frac{1}{2},\\ (c_2+d_3)c_1+c_3(c_1+c_2+d_2) &= \frac{1}{2},\\ a_1+b_1-c_1 &= 0,\\ a_2+b_2-c_2-d_2 &= 0,\\ a_3+b_3-c_3-d_3 &= 0,\\ a_3+b_2-c_3 &= 0,\\ a_2+b_1-c_2-d_3 &= 0,\\ a_1+b_3-c_1-d_2 &= 0.\end{aligned} \tag{6.29}$$

The solution to (6.29) has three free degrees of freedom. Choosing c_1, c_2, d_2 as free parameters, the other coefficients are obtained as

$$
\begin{aligned}
a_1 &= c_1 + d_2, \\
b_1 &= -d_2, \\
a_2 &= -\frac{-2d_2 - 2c_2 + 4c_1 d_2 + 2c_1 c_2 + 2c_1^2 - 2c_1 + 1}{2(d_2 + c_2)}, \\
b_2 &= \frac{2d_2^2 + 4d_2 c_2 + 4c_1 d_2 + 2c_1 c_2 + 2c_2^2 + 2c_1^2 - 2c_1 + 1 - 2d_2 - 2c_2}{2(d_2 + c_2)}, \\
b_3 &= 0, \\
c_3 &= \frac{2c_1^2 + 2c_1 d_2 - 2c_1 + 1}{2(d_2 + c_2)}, \\
d_3 &= -b_2.
\end{aligned}
\tag{6.30}
$$

For arbitrary c_1, c_2, d_2 ($c_2 + d_2 \neq 0$), the IMEX scheme (6.20) with the coefficients given in (6.30) is second-order accurate and adjoint-consistent. Among all the possible c_1, c_2, d_2, we now try to find the one that gives third-order accuracy for the convection term.

As shown in (6.23), there are several additional conditions to achieve third-order accuracy. However, there is no solution to the nonlinear system (6.30) with those additional conditions. Thus, we have to sacrifice some conditions.

To be third-order accurate for the convection term, let us consider a linear test problem,

$$\frac{\partial u}{\partial t} = \lambda u,$$

applying the explicit Runge-Kutta scheme,

$$\begin{aligned} u_n' &= u_n + h(c_1 \lambda u_n), \\ u_n'' &= u_n' + h(c_2 \lambda u_n' + d_2 \lambda u_n), \\ u_{n+1} &= u_n'' + h(c_3 \lambda u_n'' + d_3 \lambda u_n'). \end{aligned} \tag{6.31}$$

The first-, second- and third-order conditions are summarized as

$$c_1 + c_2 + c_3 + d_2 + d_3 = 1, \quad (c_2 + d_3)c_1 + c_3(c_1 + d_2 + c_2) = 1/2, \quad c_1 c_2 c_3 = 1/6. \tag{6.32}$$

Solving (6.30) together with the condition $c_1 c_2 c_3 = 1/6$ will give us third-order accuracy in the convection term. Now we can eliminate another degree of freedom as

$$d_2 = -\frac{c_2(6c_1^3 - 6c_1^2 + 3c_1 - 1)}{6c_1^2 c_2 - 1}. \tag{6.33}$$

Then,

$$b_3[(c_3(c_1 + c_2 + d_2) + c_1 d_3] + c_1 c_2 (c_3 + d_3) + c_1 c_2 b_2 = c_1 c_2 c_3 = \frac{1}{6},$$

that is, we have another condition satisfied automatically. Unfortunately, for the other three equations in the third order conditions (6.23), we cannot find any real solutions. In summary, the IMEX scheme (6.20) with the coefficients satisfying (6.30) and (6.33) is adjoint-consistent with a second-order overall accuracy and third-order accuracy for the convection term. The coefficients we choose in the simulations are:

$$\begin{aligned} &a_1 = \frac{1}{3}, \quad a_2 = -\frac{1}{2}, \quad a_3 = \frac{1}{3}, \quad b_1 = \frac{1}{6}, \quad b_2 = \frac{2}{3}, \quad b_3 = 0, \\ &c_1 = \frac{1}{2}, \qquad c_2 = \frac{1}{3}, \quad c_3 = 1, \quad d_2 = -\frac{1}{6}, \quad d_3 = -\frac{2}{3}. \end{aligned} \tag{6.34}$$

Chapter 7

The Lorenz System

As summarized in Chapter 5, the adjoint method fails in computing the shape derivatives for turbulent flows. In this chapter, we study the Lorenz system, which is a simple chaotic system, as a model problem and study its sensitivity analysis. The Lorenz system of ordinary differential equations arose from the work of Edward Lorenz, who derived it from simplified equations of convection rolls arising in the equations of the atmosphere. It is a nonlinear, three-dimensional, deterministic system of ODEs and exhibits chaotic behavior. Since 1963, the Lorenz system has been one of the most widely studied systems of chaotic ODEs because of its wide range of behaviors. Here, we are interested in the sensitivity of model-simulated quantities to changes in the parameters.

7.1 Lorenz equations

The Lorenz equations are written as

$$\begin{cases} x' = & \sigma(y-x) \\ y' = & x(r-z)-y \\ z' = & xy-\beta z \end{cases} \tag{7.1}$$

$$(x,y,z)=(x_0,y_0,z_0) \text{ at } t=0,$$

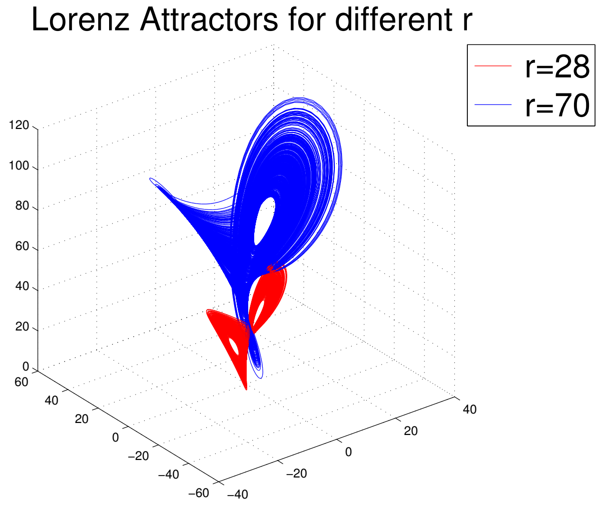

Figure 7.1: Lorenz attractors for different values of the parameter r: shown in red is the trajectory for $r = 28$, in blue for $r = 70$.

where (σ, r, β) are positive parameters, σ is interpreted as the Prandtl number, r the Rayleigh number and β is a physical proportion. The evolution of this system exhibits a strong dependence on the parameters and the initial conditions.

For different sets of parameters, we find quite different attractors. In this analysis, we fix $\sigma = 10, \beta = 8/3$ and study the sensitivity of the system with respect to the Rayleigh number r. Even for fixed σ and β, we can obtain well-separated attractors given different r. The attractors for $r = 28$ and $r = 70$ are shown in Fig. 7.1, for example. Moreover, with the same value of r, even a tiny perturbation in the initial condition can yield dramatically different trajectories. Fig. 7.2 exhibits two trajectories of the system with the same parameters starting from different initial conditions $(0.0, 1.0, 0.0)$ and $(10^{-5}, 1.0, 0.0)$. The variation of 10^{-5} in the x-coordinate leads to a clearly visible divergence of the trajectories.

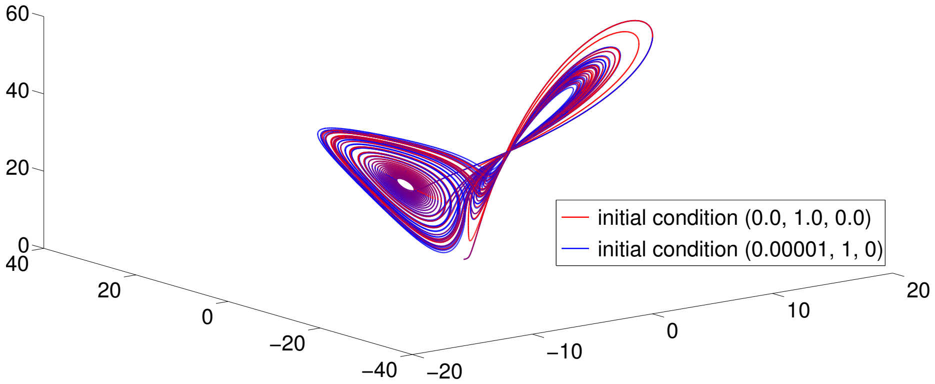

Figure 7.2: Lorenz attractors starting with initial values varying by 10^{-5} in the x coordinate with $r = 28, \sigma = 10, \beta = 8/3$. Shown in red is the trajectory for initial conditions $(0.0, 1.0, 0.0)$, in blue for the initial condition $(10^{-5}, 1.0, 0.0)$.

Below, we study the sensitivity of the system with respect to the Rayleigh number at the value $r = 28$. Fig. 7.3 show the instantaneous value of $x(t), y(t), z(t)$ for a random initial condition. The horizontal lines shows the mean values, denoted by $\langle x \rangle, \langle y \rangle, \langle z \rangle$. It may be observed that the instantaneous values of $x(t), y(t), z(t)$ display significant fluctuations and the time histories exhibit variations on a wide range of timescales. However, their mean values are more stable. Thus, for a chaotic system, we focus on the mean value rather than the time history or one particular trajectory, and the quantity of interest here is the mean value of z, which we denote by $\langle z \rangle$. Written in a mathematical framework, the sensitivity problem for the Lorenz

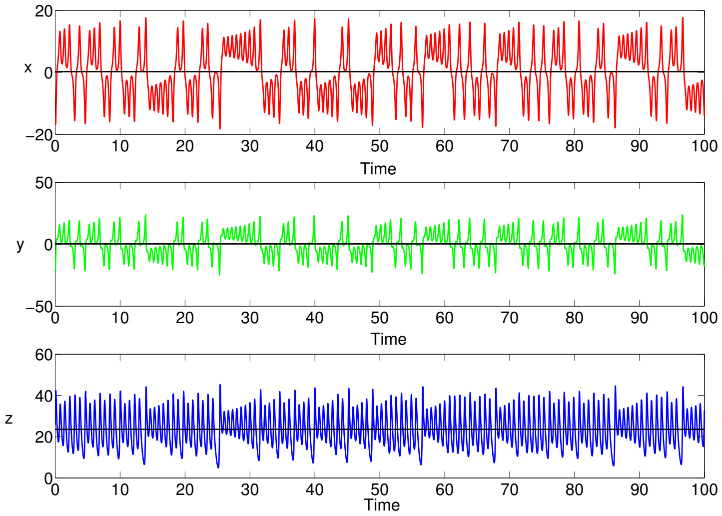

Figure 7.3: The time history of $x(t), y(t), z(t)$ and their means values.

system is: study $\frac{dJ}{dr}$, where

$$J(r) = \langle z \rangle$$

subject to

$$\begin{cases} x' &= \sigma(y-x) \\ y' &= x(r-z)-y \\ z' &= xy-\beta z \end{cases} \tag{7.2}$$

for $\sigma = 10, \beta = 8/3, r = 28$. In the sequel, we will discuss direct finite difference and adjoint approaches to compute the sensitivity, i.e., the derivative $\frac{dJ}{dr}$.

As a preparation, we recall definitions and methods used to compute mean values. For the chaotic system given in (7.1) with uncertain and random initial conditions, the solutions x, y, z are *random variables*. The *probability distribution function* (pdf) is used to characterize these random variables. Their mean values are of particular importance. The mean of z is defined by:

$$\langle z\rangle = \int_{\Omega} z\rho(t,\mathbf{s},r)\,d\mathbf{s} = \int_{\Omega_0} z(t)\rho_0\,d\mathbf{s}, \quad \text{for arbitrary } t, \tag{7.3}$$

where $\rho(t,\mathbf{s},r)$ is the pdf of (x,y,z) at time t, ρ_0 is the pdf of the initial conditions (x_0,y_0,z_0), and Ω and Ω_0 are the corresponding probability spaces. When the system is statistically stationary, $\rho(t,\mathbf{s},r)$ will converge to a stationary pdf $\bar{\rho}(r,\mathbf{s})$, which is independent in time. Thus we also have

$$\langle z\rangle := \int_{\Omega} z\bar{\rho}(\mathbf{s},r)\,d\mathbf{s}, \tag{7.4}$$

which implies that the mean is invariant in time. However, it is generally difficult to compute the mean directly from (7.3) or (7.4) because ρ and $\bar{\rho}$ are not available and computationally expensive to determine. Instead, we use different averaging methods to approximate the mean, such as ensemble and time averages.

The ensemble average is the average over a large number of realizations of the system with different initial conditions, i.e.,

$$\langle z\rangle_N = \frac{1}{N}\sum_{n=1}^{N} z_n(t), \qquad \text{for arbitrary fixed } t,$$

where $z_n(t)$ is the trajectory of z on the nth realization. The ensemble average $\langle z\rangle_N$ serves as a good approximation of the mean because it is sampling the distribution of the initial conditions ρ_0 using different realizations. Thus, $\langle z\rangle_N$ converges to $\langle z\rangle$ as the sample size N increases. The convergence rate depends on the sampling method and sample size.

T	$h = 10^{-3}$	$h = 10^{-4}$	$h = 10^{-5}$	$h = 10^{-6}$
$T = 10$	23.298823	23.398594	23.305425	23.238065
$T = 10^2$	23.653910	23.623302	23.615610	23.539938
$T = 10^3$	23.552515	23.516541	23.542850	23.543939
$T = 10^4$	23.551697	23.546635	23.542468	23.544779
$T = 10^5$	23.549979	23.552049	23.550768	23.550485
$T = 10^6$	23.549979	23.552219	23.549643	23.550886
$T = 10^7$	23.550106	23.550967	23.549960	23.550180
$T = 10^8$	23.550027	23.549853	23.549929	*
$T = 10^9$	23.549892	23.549853	*	*

Table 7.1: The converges of $\hat{z}_T$ for different averaging time T and time step size h used in RK4 (The "*" denotes that the data has not been computed).

For a statistically steady system under the assumption of ergodicity, the long-time average can also be employed as an approximation of the mean, i.e.,

$$\hat{z}_T = \frac{1}{T}\int_0^T z(t)dt \approx \langle z \rangle,$$

where $z(t)$ is the instantaneous solution of the ODE system (7.1) in a single realization with an arbitrary given initial condition. As shown in 5.1, if the averaging time T is taken large enough, the time average $\hat{z}_T$ converges to the mean as its standard deviation decreases to 0 with order $T^{-1/2}$.

To numerically verify the convergence of the time average, we solve the Lorenz system (7.1) using the classical fourth-order Runge-Kutta method (RK4) and compute $\hat{z}_T$ for different averaging times T and time step size h. The results are shown in Table 7.1. We can conclude that, as we increase the averaging time, the time-average converges. However, this convergence is very slow: there are at most 4 or 5 accurate digits in the averaging results $\hat{z}_T$ for $T = 10^8$.

T	$r = 28$	$\delta r = -0.1$	$\delta r = 0.1$	$\delta r = -0.01$	$\delta r = 0.01$
10^3	23.516561	23.465705	23.634621	23.537455	23.591763
10^4	23.546635	23.456458	23.644350	23.542639	23.579644
10^5	23.552049	23.446437	23.649158	23.538276	23.565785
10^6	23.552219	23.448701	23.650288	23.539326	23.564301
10^7	23.550967	23.448080	23.650349	23.538423	23.565327
10^8	23.549853	23.449751	23.650209	23.539936	23.559916

Table 7.2: $\hat{z}_T$ for different averaging time T and δr with $h = 10^{-4}$.

7.2 Sensitivity analysis

With the preparation work for the mean values computation, now we discuss different methods used in the sensitivity analysis of the Lorenz system to compute $\frac{d\langle z\rangle}{dr}$ as given in (7.2). We start with the direct finite difference method.

7.2.1 Direct finite difference approach

Based on averaging methods to approximate $\langle z\rangle$, we can use finite differences to compute the sensitivity $\frac{dJ}{dr}$ directly. Note that the derivative of $\langle z\rangle$ with respect to the parameter r is defined by

$$\frac{d}{dr}\langle z\rangle = \lim_{\delta r \to 0} \frac{\langle z(r+\delta r)\rangle - \langle z(r)\rangle}{\delta r}, \tag{7.5}$$

where δr is a perturbation of the parameter r. Results of $\hat{z}_T(r+\delta r)$ with different T and δr are summarized in Table 7.2.

As we have shown earlier, $\hat{z}_T$ converges to $\langle z\rangle$ with order $T^{-1/2}$, and we have at most 4 or 5 accurate digits in $\hat{z}_T$ given $T = 10^8$, thus the averaging error is of order 0.001. Denoting the time averaging error as ε_T,

$$\frac{\hat{z}_T(r+\delta r) - \hat{z}_T(r)}{\delta r} = \frac{\langle z(r+\delta r)\rangle - \langle z(r)\rangle + 2\varepsilon_T}{\delta r} = \frac{d}{dr}\langle z\rangle + o(\delta r) + \frac{2\varepsilon_T}{\delta r},$$

T	$\delta r = 0.1$ (forward)	$\delta r = 0.1$ (center)	$\delta r = 0.01$ (forward)	$\delta r = 0.01$ (center)	$\delta r = 0.001$ (forward)	$\delta r = 0.001$ (center)
10^3	1.180	0.6303	7.5	2.7154	27.264	-5.343
10^4	0.977	0.939	3.291	1.845	2.789	4.138
10^5	0.969	1.014	1.374	1.375	-3.559	-0.180
10^6	0.981	1.001	1.201	1.239	-0.981	0.018
10^7	0.994	1.011	1.436	1.345	-0.192	0.081
10^8	1.004	1.002	1.006	0.999	0.108	0.101

Table 7.3: Finite difference derivatives for different T and δr using forward and centered difference methods.

where $o(\delta r)$ is the finite difference error. The dominant error, the term $\frac{2\varepsilon_T}{\delta r}$, is determined by the averaging error and the perturbation size. Thus, we cannot choose a too small δr since the time averaging error will be magnified by a factor of $\frac{1}{\delta r}$. The finite difference gradients computed with different T and δr are shown in Table 7.3. We can see that, for $\delta r = 0.1$ and $\delta r = 0.01$, the finite difference derivatives converge to approximately 1 as the averaging time T increases. This suggests that $\frac{d}{dr}\langle z \rangle$ exists and is approximately 1. However, for $\delta r = 0.001$, the finite difference result is unrealizable because of the error amplification factor $\frac{1}{0.001} = 1000$, which induces a significant error.

In summary, the finite difference method gives the approximate derivatives by solving the model with the perturbed input. It suggests the differentiability of the objective function. However, the main disadvantage is that this approach is not scalable: if the sensitivity to several model parameters is required, a separate model solve must be performed for each parameter, which is computationally expensive.

7.2.2 The adjoint approach

An alternative method to obtain the sensitivity is to use the adjoint method (also called the Lagrange method). We refer to section 2.2 for a brief introduction. The advantage of the adjoint approach over the finite difference method is that it gives the sensitivity of a single component of the model output to *all* the model input variables simultaneously. Thus, for a problem with many parameters, it is computationally more efficient.

Recalling the sensitivity problem (7.2) and using the Lagrange method, we can conclude the adjoint system of the Lorenz equations for fixed final time T,

$$\begin{cases} -\lambda' &= -\sigma\lambda + \gamma y + \alpha(\rho - z) \\ -\alpha' &= \sigma\lambda - \alpha + \gamma x \\ -\gamma' &= -\alpha x - \beta\gamma - 1 \end{cases} \tag{7.6}$$

$$(\lambda, \alpha, \gamma) = (0,0,0) \quad \text{at } t = T,$$

and the incremental state equations:

$$\begin{cases} \delta x' &= \sigma(\delta y - \delta x) \\ \delta y' &= \delta x(r - z) + x(\delta r - \delta z) - \delta y \\ \delta z' &= \delta x y + x\delta y - \beta\delta z \end{cases} \tag{7.7}$$

$$(\delta x, \delta y, \delta z) = (0,0,0) \quad \text{at } t = 0,$$

where λ, α, γ are the adjoint variables, $\delta x, \delta y, \delta z$ are the incremental state variables, and δr the variation in r. The adjoint and incremental variables correspond to the changes of the state variable for an infinitesimal perturbation δr, i.e., $\frac{dz}{dr} = \delta z$. The first-order derivative is given by:

$$\frac{d}{dr}\langle z\rangle = \langle\frac{dz}{dr}\rangle = \langle\delta z\rangle. \tag{7.8}$$

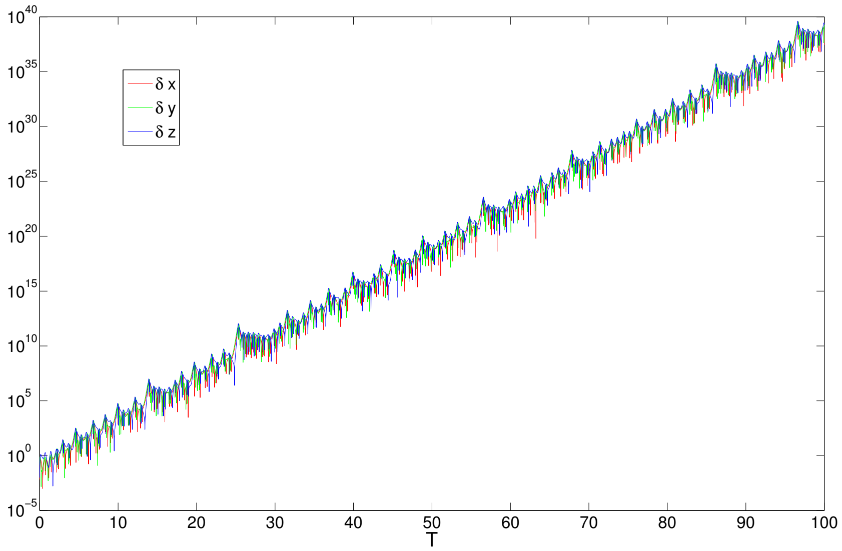

Figure 7.4: Exponential growth for the incremental state variables.

Here, we use the fact that the mean operator commutes with the differentiation operator. Next, we solve the incremental system for δz and compute its time average to approximate $\frac{d}{dr}\langle z \rangle$.

However, due to its chaotic nature, the Lorenz system is very sensitive to small perturbations, i.e., a tiny perturbation leads to a divergence in the trajectories. Thus, the incremental variables, which describe this sensitivity, grow rapidly in time. The solutions $\delta x, \delta y, \delta z$, which linearly depend on δr, are shown in Fig. 7.4 with $\delta r = 1$. As seen from the plot, the magnitude of the incremental variables grows exponentially. The average exponential growth rate here is about 0.9, which is consistent with the first Lyapunov exponents of the system with the given parameters values (see [72]).

Now, to get the derivative, we compute the mean value of δz by its time average. However, the rapid growth for the incremental variables makes the time average meaningless since

$$\hat{\delta z}_T := \frac{1}{T}\int_0^T \delta z\, dt \to \infty,$$

that is, $\hat{\delta z}_T$ is unbounded. Thus, the direct application of the adjoint approach fails for chaotic systems.

As a remedy, in [45] the authors propose the *ensemble adjoint approach*, which is to use the ensemble average rather than time average to approximate the mean value of δz. This approach estimates the sensitivity by taking ensemble averages of the adjoint gradient developed within a short time, i.e.,

$$\frac{d}{dr}\langle z\rangle \approx \frac{1}{N}\sum_n \int_0^T \delta z_n\, dt, \tag{7.9}$$

where δz_n is the nth realization of the incremental state system, and T is the developing and averaging time for each realization. After comparing the results from (7.9) with short, intermediate, long and very long integration time T, they suggest that the ensemble-averaged results using intermediate integration time ($T \approx 0.44$) gives the most reasonable estimate.

However, the incremental system given in (7.7) describes the sensitivity of the state variables for a short time interval. Its time evolution always starts from the initial condition $(0,0,0)$. To capture the full information of the current state, an appropriate amount of development time is needed, and T cannot be too small. On the other hand, we cannot choose the integration time T too long to avoid a large δz_n given the average growth rate $e^{0.9t}$ (which is the first Lyapunov exponent).

T	adjoint derivative	finite difference $\delta r = 0.01$	finite difference $\delta r = 10^{-4}$	finite difference $\delta r = 10^{-6}$
2	2.0637569	2.0637574	2.0637569	2.0637572
4	4.5204501	4.5204496	4.5204501	4.5204501
6	6.8750610	6.8750877	6.8750610	6.8750620
8	6.6792221	6.6792521	6.6792221	6.6792226
10	15.4216633	15.4242134	15.4216633	15.4216671
12	-10.0729561	-9.4950218	-10.0728989	-10.0729828
14	430.64407	-0.9606939	430.94662	430.64456
16	-2541.2749	39.1021042	-2553.1785	-2541.2788

Table 7.4: Comparison of finite difference derivatives and adjoint derivatives for different T and δr for the sample starting with $(x_0, y_0, z_0) = (3.149719, 1.143445, 3.588247)$.

Otherwise, a large number of samples will be needed for the ensemble averaging due to the slow convergence rate and the large magnitude of δz_n. Also, the rapid propagation of the numerical errors will deteriorate the derivatives computed using the adjoint approach.

To find an appropriate development time T, we perform some numerical tests to find out how the numerical errors grow with T by comparing the adjoint gradients and finite difference gradients at different T. That is, with the equality

$$\langle\frac{z(r+\delta r)-z(r)}{\delta r}\rangle = \frac{\langle z(r+\delta r)\rangle - \langle z(r)\rangle}{\delta r} = \langle \delta z\rangle, \tag{7.10}$$

and the ensemble average method,

$$\langle\frac{z(r+\delta r)-z(r)}{r}\rangle \approx \frac{1}{N}\sum_n \frac{z_n(r+\delta r)-z_n(r)}{r}, \quad \langle \delta z\rangle \approx \frac{1}{N}\sum_n \delta z_n = \langle \delta z\rangle_N,$$

we study $\frac{z_n(r+\delta r)-z_n(r)}{r}$ with δz_n for a single sample as the integration time T changes. The results of the comparison are listed in Tables 7.4 and 7.5, for two different samples.

T	adjoint derivative	finite difference $\delta r = 0.01$	finite difference $\delta r = 10^{-4}$	finite difference $\delta r = 10^{-6}$
1	0.8046634	0.8046650	0.8046634	0.8046634
2	2.3939571	2.3940473	2.3939571	2.3939569
3	5.4613957	5.4618783	5.4613957	5.4613957
4	0.2996249	0.2963130	0.2996246	0.2996253
5	-1.0055308	-0.9423273	-1.0055245	-1.0055305
6	47.2752419	48.8699684	47.2754021	47.2752342
7	-6.5140357	19.6534023	-4.7699199	-6.5138569
8	-10535876	8.9092855	-21.3869381	-424708.69

Table 7.5: Comparison of finite difference derivatives and adjoint derivatives for different T and δr for the sample starting with $(x_0, y_0, z_0) = (9.1461, 3.8432, 33.3986)$.

For some samples, as shown in Table 7.4, the two results match very well even at $T = 14$. However, for some samples, we observe an apparent divergence of these derivatives as early as $T \geq 6$ (Table 7.5). This is caused by the high sensitivity of the chaotic system and the propagation of round-off errors in numerical computation. In our numerical test of the ensemble adjoint approach, we test the cases of $T = 8$, $T = 6$ and $T = 4$, which give $\delta z \sim 10^3$ with the average growth factor.

To draw the samples for the adjoint ensembling, instead of sampling the initial conditions to get different realizations of the system, we can get independent samples from one realization by virtue of the ergodic hypothesis. That is, given the knowledge that the average growth rate of the incremental solutions is about $e^{0.9t}$, two points in one realization, which are separated by 40 time units ($e^{0.9 \times 40} \sim 10^{15}$), can be viewed and used as start points for two independent samples since the accumulation of round-off errors makes the correlation of any of those two points negligible.

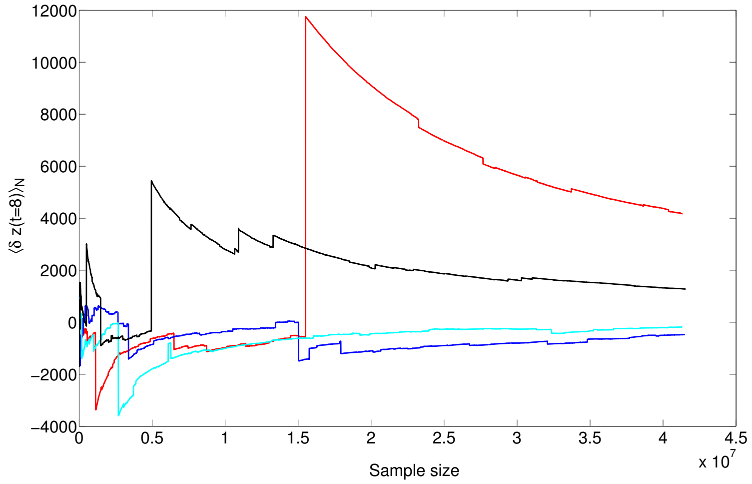

Figure 7.5: Convergence of the ensemble averaged results for $T = 8$, i.e., $\langle \delta z(t = 8) \rangle_N$, colors denote independent realization with different initial conditions.

Thus, within a realization of the Lorenz system, we take each time interval with a length of 40 time units as a sample. For each sample, using zero initial conditions for the incremental state equations and the state solutions from the current realization, we solve the incremental system for $\delta z_n(t = T)$ with different T. The ensemble average of $\delta z_n(t = T)$ gives us a new approximation of $\langle \delta z \rangle$. The numerical results of $\langle \delta z \rangle_N$ from different realizations are shown in Figures 7.5 and 7.6, for different T. Recall that the finite difference results suggest that the derivative is about 1. So based on the equality (7.10), $\langle \delta z \rangle_N$ is expected to converge to 1 approximately as we increase the sample size N. From Fig. 7.5, we see that, for each individual run, the averaged results seem to converge, but the convergence is too slow to indicate the values they are converging to with current sample sizes.

There are several causes for the slow convergence of $\langle \delta z \rangle_N$ as shown in Fig. 7.5. An important one is that the convergence rate for the ensemble average is

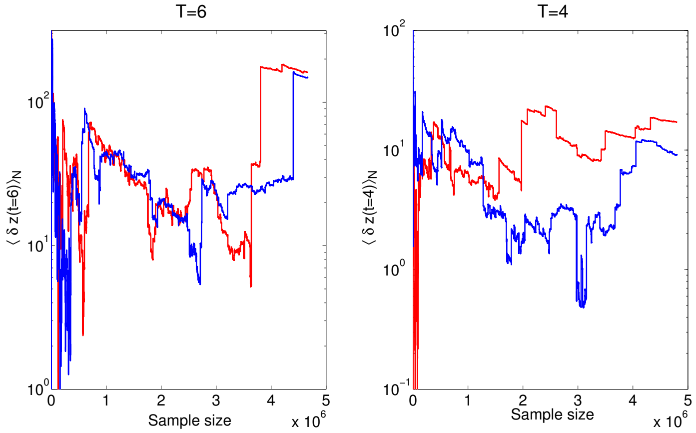

Figure 7.6: Convergence of the ensemble averaged results for $T = 4$ and $T = 6$, colors denote independent realization with different initial conditions.

of the order of $N^{-1/2}$. Thus, we need to increase the sample size N by a factor of 100 to obtain one more accurate digit in the ensemble averaging results. Another factor is the existence of the sharp-changing points in the results curves. These sharp points correspond to those sample points, whose incremental systems have much larger growth factors than the average value. For example, for the average growth rate $e^{0.9t}$ (see previous discussion about the first Lyapunov exponents and [72]), $\delta z \sim 10^3$ at $T = 8$. However, there are points on the trajectories which give $\delta z \sim 10^9$ at the end of the same integration time. This extreme variability is also shown in the distribution of the growth factors for each sample in Fig. 7.7. Let λ_g denote the local growth factor. While the mean of the growth factors computed from each sample is about 0.9 in this run, which gives $\delta z \sim 10^3$, the maximum value λ_g is about 2.68, which leads to $\delta z \sim 10^9$. Thus, starting from these points, a small perturbation in the parameter r will lead to rapid separation of the trajectories. Fig. 7.2.2 shows the comparison of the time histories at different time t for $\delta r = 0.001$ with two different

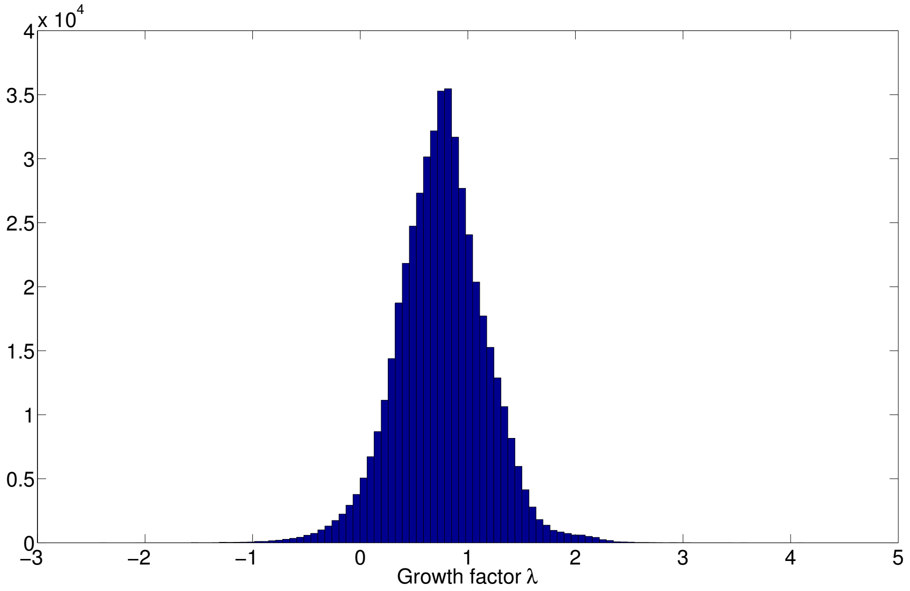

Figure 7.7: Distribution of the growth factor for the incremental systems.

starting points.

For the first case shown on the top row, the trajectories for $r = 28$ and $r = 28.0001$ are almost on top of each other at $T = 8$; however, for the second one with $(x_0, y_0, z_0) = (9.146, 3.843, 33.40)$, the trajectory for $r = 28.0001$ is diverging away from the curve correspond to $r = 28$ rapidly. In summary, due to the existence of these highly sensitive points, the ensemble averaged derivatives converge very slowly to the correct derivative. In the sequel, we discuss possible remedies.

7.2.3 Damping factors

In the previous section we concluded that the direct application of the adjoint method to compute the sensitivity of Lorenz system fails because of its chaotic properties, as a consequence, rapid growth of the incremental systems. We next re-

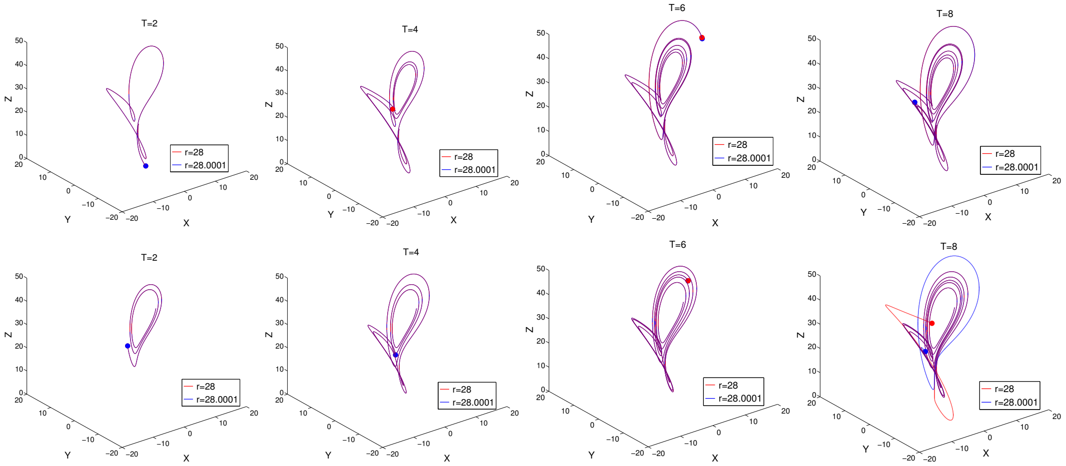

Figure 7.8: Snapshots of the trajectory of the Lorenz system with $r = 28$ (shown in red) and $r = 28.0001$ (shown in blue) with different starting points: top row: $(x_0, y_0, z_0) = (-0.5696, -1.5496, 18.7478)$, bottom row:$(x_0, y_0, z_0) = (9.146, 3.843, 33.40)$.

port on our experiments trying to overcome this issue.

We recall that the quantity of interest is

$$J(r) = \langle z(r) \rangle = \int_{\Omega_0} \rho_0(s) z(t, r, s)\, ds, \tag{7.11}$$

where $\rho_0(s)$ is the pdf of the initial condition and Ω_0 is the corresponding probability space. With the assumption that the system is statistically steady and ergodic, we have

$$\langle z \rangle = \frac{1}{T}\int_0^T \langle z \rangle\, dt = \frac{1}{T}\int_0^T \Big(\int_{\Omega_0} \rho_0(s) z(r,s,t)\, ds\Big)\, dt = \frac{1}{T}\int_{\Omega_0} \rho_0(s)\Big(\int_0^T z(r,s,t)\, dt\Big)\, ds.$$

Using the stationarity property of the system, we can also rewrite (7.11) as

$$J(r) = c\int_0^T e^{-\alpha t} \langle z \rangle\, dt = c\int_{\Omega_0} \rho_0(s)\Big(\int_0^T e^{-\alpha t} z(r,s,t)\, dt\Big)\, ds := J_\alpha(r), \tag{7.12}$$

where $\alpha > 0$ is a damping factor, $c = \frac{\alpha}{1-e^{-\alpha T}}$ is the normalization constant. Notice that for arbitrary α,

$$J_\alpha(r) \equiv J(r) = \langle z \rangle. \tag{7.13}$$

The sensitivity of the new objective functional $J_\alpha(r)$ is given by:

$$\frac{d}{dr} J_\alpha = c \int_{\Omega_0} \rho_0(s) (\int_0^T e^{-\alpha t} \delta z \, dt) \, ds \approx \frac{c}{N} \sum_n (\int_0^T e^{-\alpha t} \delta z_n \, dt). \tag{7.14}$$

Here we use the ensemble average again to sample the distribution of the initial conditions to approximate the mean operator. By rewriting the $\langle z \rangle$ in the equivalent form J_α and choosing an appropriate α to compensate for the exponential growth of δr, we hope to achieve better convergent results of the sensitivity. Notice that for $\alpha = 0$ (and $c = 0$), this simplifies to the ensemble adjoint approach.

We compute the ensemble adjoint averages for different damping factors α and integration times T. The results are shown in Fig 7.9. By introducing the damping factor, we achieve a more reasonable estimation for a moderate or large integration time T. Equation (7.13) implies that the ensemble adjoint average computed from (7.14) should be independent of the value of α. However, this is not true for the computed numerical results as shown in Fig 7.9. It is shown that, if α is chosen to be equivalent to the growth factor of δz_n, we obtain the optimal ensemble average results. In general, the growth factor is not known or expensive to compute for more complicate systems. The results shown in the figure are also affected by the slow convergence of the ensemble methods. Thus, applying this method to chaotic systems, for example, turbulent flows, is not promising.

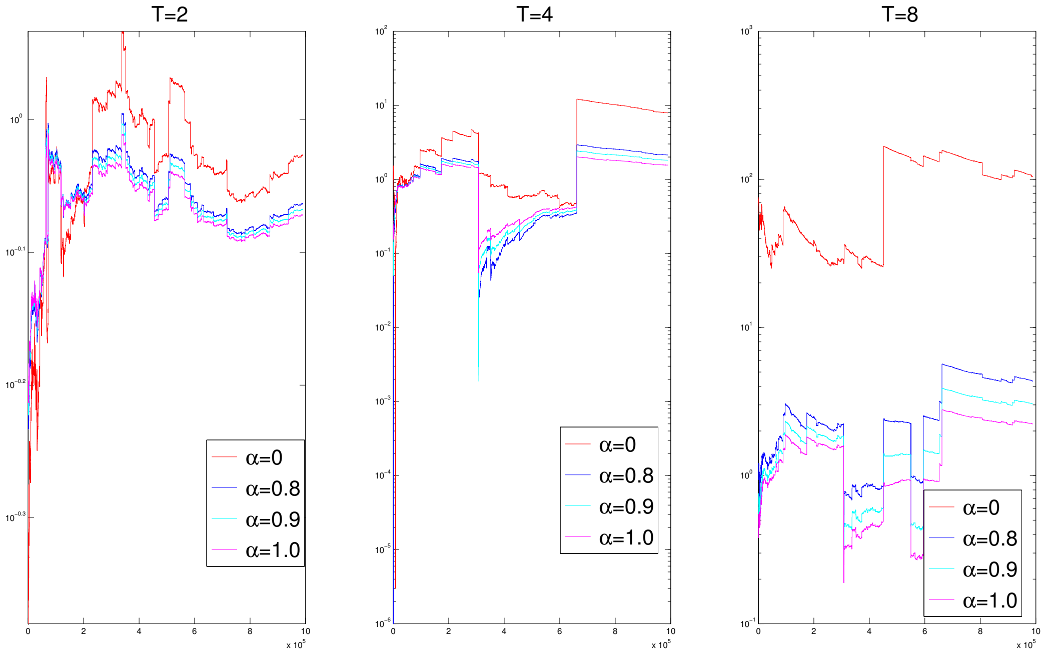

Figure 7.9: Convergence of the ensemble averaged results with different damping factors α (shown in plot: $\alpha = 0, 0.8, 0.9$ and 1.0) and averaging time T (left, $T = 2$; middle $T = 4$; right $T = 8$), here x-axis denotes the size of ensemble samples and y-axis denotes $\frac{dJ_\alpha}{dr}$.

7.2.4 Mean of Lorenz equations

The methods discussed above are all based on the fact that

$$\frac{d}{dr}\langle z\rangle = \langle\frac{dz}{dr}\rangle = \langle\delta z\rangle.$$

However, it is difficult to get a convergent mean value of $\frac{dz}{dr}$ by averaging the incremental solution δz. Now we seek to get $\langle\delta z\rangle$ directly from the system or find an alternative expression that enables us to get better convergent results. For a statistically steady system, we have $\langle x'\rangle = 0$ (here, the prime denotes the time derivative), thus the mean of the Lorenz equations becomes:

$$\begin{cases} \langle y\rangle - \langle x\rangle & = 0 \\ \sigma\langle x\rangle - \langle xz\rangle - \langle y\rangle & = 0 \\ \langle xy\rangle - \beta\langle z\rangle & = 0. \end{cases} \tag{7.15}$$

Notice that the Lorenz system is symmetric in x and y, we have

$$\langle x\rangle = \langle y\rangle = 0, \qquad \langle xz\rangle = 0.$$

From the last equation in (7.15), which implies:

$$\langle xy\rangle = \beta\langle z\rangle, \tag{7.16}$$

thus

$$\langle\delta z\rangle = \frac{d}{dr}\langle z\rangle = \frac{1}{\beta}\frac{d}{dr}\langle xy\rangle = \frac{1}{\beta}\langle\delta xy + x\delta y\rangle. \tag{7.17}$$

That is, $(\delta xy + x\delta y)$ also describes the sensitivity of Lorenz system with respect to the parameter r. Thus, instead of using ensemble average or time average of δz, we employ the equivalent expression $\delta xy + x\delta y$. The time average results of the

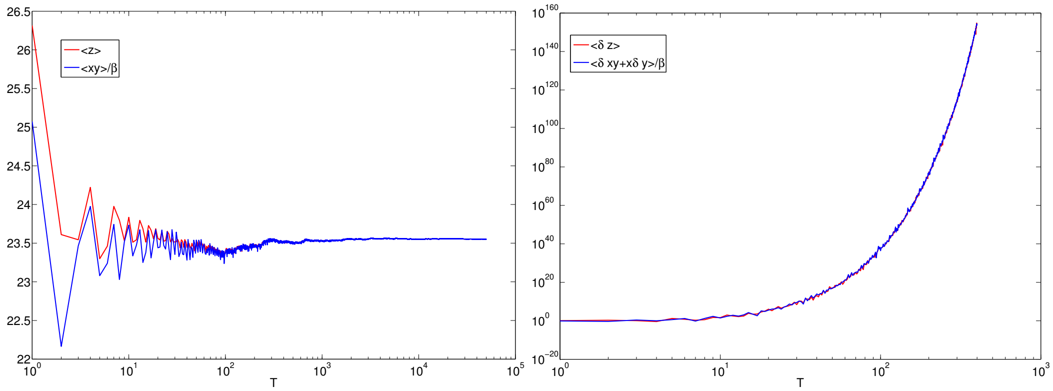

Figure 7.10: Comparison of different time averages: left: $\langle z\rangle_N$ with $\langle xy\rangle_N/\beta$; right, $\langle\delta z\rangle_N$ with $\langle\delta xy+x\delta y\rangle_N/\beta$.

two expressions are compared in Fig. 7.10. Unfortunately, the convergence results do not improve by switching to the form of $(\delta xy+x\delta y)$. In some tests, archiving convergence for $\langle\delta xy+x\delta y\rangle_N/\beta$ appears even harder than for δz.

7.2.5 pdf approach

An alternative expression for $\frac{\partial\langle z\rangle}{\partial r}$ is:

$$\frac{\partial\langle z\rangle}{\partial r}=\frac{\partial}{\partial r}\left(\int_{\Omega} z\bar{\rho}(z;r)\,dz\right)=\int_{\Omega} z\frac{\partial\bar{\rho}}{\partial r}\,dz. \tag{7.18}$$

Here, $\bar{\rho}(z;r)$ is the stationary *probability distribution function* (pdf) of z for parameter r, Ω is the support of the sample space, i.e,

$$\Omega=\{z:\bar{\rho}(z;r)>0\}.$$

If we can compute $\frac{\partial\bar{\rho}}{\partial r}$, (7.18) can be used as an alternative method for our sensitivity analysis. To compute $\frac{\partial\bar{\rho}}{\partial r}$, let us first derive the pdf transport equation.

Consider the abstract form of a chaotic system:

$$\begin{aligned}\frac{\partial x}{\partial t} &= N(x,r),\\ x = x_0 &\quad \text{at } t=0,\end{aligned} \tag{7.19}$$

where the initial condition x_0 is a random variable with pdf $\rho(x_0)$. We use $\rho(s,r;t) = \rho(x(t),r)$ to denote the pdf of random variable x at time t at location s with given parameter r. For a given realization of the system, the (one-point, one-time, Eulerian) fine-grained pdf (see [59]) of x is defined by

$$\rho_x'(s,r;t) = \delta(x(t,r)-s), \tag{7.20}$$

where s is the sample-space variable. $\rho_x'(s,r;t)$ has the following properties:

$$\begin{aligned}\langle \rho_x'(s,r;t)\rangle &= \rho(s,r;t),\\ \langle \phi(s;t)\rho_x'(s,r;t)\rangle &= \langle \phi(s;t)|x(t)=s\rangle \rho(s,r;t),\end{aligned} \tag{7.21}$$

where $\psi(s;t)$ is a random field. Differentiating (7.20) with respect to t using the chain rule, we obtain

$$\begin{aligned}\frac{\partial}{\partial t}\rho_x'(s,r;t) = \frac{\partial}{\partial x}(\delta(x(t,r)-s))\frac{\partial x}{\partial t} &= -\frac{\partial \rho_x'(s,r;t)}{\partial s}\frac{\partial x}{\partial t}\\ = -\frac{\partial}{\partial s}\left(\rho_x'(s,r;t)\frac{\partial x}{\partial t}\right)&,\end{aligned} \tag{7.22}$$

whose mean yields,

$$\frac{\partial \rho(s,r;t)}{\partial t} = -\frac{\partial}{\partial s}\Big(\rho(s,r;t)N(s,r)\Big). \tag{7.23}$$

When the chaotic system becomes statistically stationary after a development time, the pdf $\rho(x(t),r)$ converges to a time-independent function $\bar{\rho}(x,r)$, and thus we have

$$\frac{\partial}{\partial s}\Big(\bar{\rho}(s,r)N(s,r)\Big) = 0 \qquad \text{or} \qquad \nabla\cdot(\bar{\rho}N) = 0. \tag{7.24}$$

This defines the transport equation for the pdf $\bar{\rho}$. For a detailed derivation, see for instance [59].

Since $\bar{\rho} > 0$ on Ω, let us define a new variable f as

$$\bar{\rho}(s,r) = e^{f(s,r)}.$$

Then,

$$\frac{\partial \bar{\rho}}{\partial r} = e^{f(s,r)} \frac{\partial f}{\partial r},$$

and thus, we only need to compute $\frac{\partial f}{\partial r}$ and $\bar{\rho}(s,r)$ to get $\frac{\partial \langle z \rangle}{\partial r}$. Using the new variable f, equation (7.24) can be written as

$$\frac{\partial f}{\partial s} \cdot N + \nabla \cdot N = 0. \tag{7.25}$$

Derivatives with respect to r on (7.25) gives

$$\frac{\partial^2 f}{\partial s \partial r} \cdot N + \frac{\partial f}{\partial s} \cdot \frac{\partial N}{\partial r} + \nabla \cdot (\frac{\partial N}{\partial r}) = 0. \tag{7.26}$$

Notice that integrating $\frac{\partial^2 f}{\partial s \partial r}$ along the boundary $\partial \Omega$ yields $\frac{\partial f}{\partial r}$. The the challenge for computing the sensitivity becomes computing $\frac{\partial f}{\partial s} \cdot \frac{\partial N}{\partial r}$. However, we found no way to compute this term directly. Again, the pdf method hits a wall.

To summarize, in this chapter, we studied the Lorenz system and experimented with different approaches to study its sensitivity with respect to the model parameter r. The finite difference method, adjoint method, ensemble adjoint method, the damping factor method and the pdf method are explored and some numerical results are presented. Due to the high sensitivity of chaotic systems and the resulting exponential growth of adjoint and incremental systems, the numerical results from each of these methods are unsatisfactory.

Chapter 8

Discussion and future work

In this dissertation, the effects of roughness topography on fluid behavior, in particular, the drag acting on a fluid, are studied using tools from shape calculus. Prediction of the effects of roughness on fluid flows is a long-standing problem in the engineering of fluid systems, which has significant consequences for the design and operation of many such systems. The primary flow situations of interest here are those in which roughness has order-one effects: hydrodynamic lubrication and high Reynolds number turbulence. They have broad applications in important areas such as transportation, propulsion and energy systems.

For the lubrication case, we studied the drag of laminar flow, i.e., the flow satisfies the stationary Navier-Stokes equations. Chapters 3 and 4 contain several of the fundamental contributions of this dissertation. There, we derived the shape gradient and shape Hessian. The analysis shows that the shape gradient is zero at a flat boundary, and thus the shape Hessian provides the lowest-order description of the roughness effect on the roughness functional $\mathscr{D}$. The homogeneity and translation invariance of the Hessian operator imply that the Fourier functions are the eigenfunctions of the shape Hessian. This allows the shape Hessian to be completely characterized by its eigenvalue spectrum. For Stokes flows, the analytical eigenvalues are all positive and increase linearly with wavenumber. Thus, the flat boundary corresponds to a global minimum of the roughness functional and the Hessian is a pseudo-differential operator of order 1. For Navier-Stokes flows, the flat boundary also corresponds to a stationary point of $\mathscr{D}$. The sensitivity of the roughness functional to Fourier basis perturbations increases with the Reynolds number. The ratio of the Navier-Stokes shape Hessian eigenvalues to those for Stokes flow depends only on the characteristic Reynolds number Re^* (the Reynolds number based of the scale height of the incremental solutions), as shown in Fig 4.2.

The simple structure of the shape Hessian for laminar flows allows the perturbation $\delta\mathscr{D}$ to a general small-amplitude roughness to be approximated easily from the roughness spectrum,

$$\delta\mathscr{D} \approx S^2 \nu \sum_k (\alpha_k^2 + \beta_k^2)\gamma(k, \mathrm{Re}),$$

where S is shear rate (the ratio of velocity on the top wall with the height of the channel), $\gamma(k, \mathrm{Re})$ is the non-dimensional eigenvalue with wavenumber k and Reynolds number Re, α_k, β_k are the Fourier coefficients of the expansion of the roughness topography (the shape perturbation). Results of a comparison of the prediction with the real dissipation (given in Figs 4.3 and 4.4) show that the prediction describes the behavior of the roughness functional accurately for small roughness height kh ($kh < 10^{-2}$ in our test), and it degrades as kh and the Reynolds number get larger. This degradation happens faster for larger Reynolds numbers. Moreover, the prediction error grows quadratically with kh. While only valid for small roughness heights, this prediction is nonetheless a useful result as it provides a simple metric for the effect of roughness surface topography on the drag.

A possible application of the explicit knowledge of the shape Hessian spectrum, which is not discussed in this dissertation, is to construct a preconditioner for shape optimization problems. In recent years, a number of numerical methods for large-scale PDE-constrained optimization have emerged (see the review in [3]). Among these are the modern inexact Newton-conjugate gradient optimization methods that, like the matrix-free eigensolvers, require only the action of the Hessian on a vector, and thus only a single pair of incremental state/adjoint Navier-Stokes solves per CG iteration. Preconditioning then becomes important to speed up CG iterations. Our explicit knowledge of the structure of the flat-boundary shape Hessian for Stokes flow (as well as Navier-Stokes flow) might be useful to precondition the CG iterations for curved boundary.

In chapter 5, we presented our efforts to carry out the roughness analysis of turbulent flows. Because of the random nature of turbulence, the quantity of interest is the mean of the drag acting on the flows. We approximate the mean by a time-average, use the adjoint method to derive shape derivatives and compute them using direct numerical simulation. However, due to the exponential growth of the adjoint and incremental variables, the time-averaged gradients are unbounded. This was also illustrated for the Lorenz system in Chapter 7. Thus, the adjoint method is not directly applicable for chaotic systems. These challenges in the sensitivity analysis for the Lorenz system and turbulence are not particular to our study. Instead, they are common phenomena in the analysis of chaotic systems, for instance, in ocean circulation models and goal-oriented error estimation. As pointed out in [38, 39, 41, 42, 44, 52, 65], since these models obey chaotic dynamics with limited predictability, the direct application of the adjoint method to these models is problematic.

For the sensitivity analysis of the Lorenz system, we first computed the derivative of the objective quantity J with respect to the model parameter r using the finite difference

$$\frac{dJ}{dr} \approx \frac{\langle z(r+\delta r)\rangle - \langle z(r)\rangle}{\delta r}, \quad \text{where } J = \langle z(r)\rangle \text{ and } \delta r \text{ is small.}$$

The numerical results show that the finite difference derivative converges, which suggest that the desired sensitivity (dJ/dr) exists and is finite. However, the finite difference approach is not computationally scalable and becomes very expensive if several model parameters are considered. Thus, it is impractical for our shape sensitivity analysis of turbulence since the shape perturbation we consider has a large dimension.

The adjoint method (or more generally, the tangent linear method), which gives the sensitivity to all parameters simultaneously, is based on

$$\frac{dJ}{dr} \approx \langle \frac{z(r+\delta r) - z(r)}{\delta r} \rangle.$$

While $\langle z(r+\delta r)\rangle$ varies smoothly with δr, $\delta z(\delta r,t) = z(r+\delta r,t) - z(r,t)$ is very sensitivity to infinitesimal perturbation δr, and results in an exponential growth of its magnitude with time. The time averaging used to approximate the mean operator has been restricted to a very short time horizon, which is too short to allow the adjoint system to be fully developed, and after this short time period, the adjoint is unstable. Thus, the stability and predictability of the adjoint method are very limited. As remedies to the shortcomings of the adjoint approach, other methods (for instance, the ensemble adjoint method, the damping factor method) were studied and tested for the Lorenz system. Instead of using the average over a long time history, we take ensemble average of large samples to approximate the mean. Unfortunately, for the same reason, each sample is restricted to a short time region, and thus the results of the ensemble gradient were not satisfactory.

In summary, despite the complexity and challenge in the study of chaotic systems, their sensitivity analysis is very important and has attracted more and more attention. The adjoint method describes the detailed sensitivity of the systems and can be used in problems with finite time horizon. However, it cannot capture the sensitivity of the statistical quantities, which are often of interest in chaotic systems. Finding a new approach to capture and understand the sensitivity of the statistics will be very useful and remains an open problem.

Appendices

Appendix A

Numerical Algorithms for forwards systems (state and incremental state equations)

As stated in Chapter 5, solving the state, adjoint and incremental equations numerically is essential for the sensitivity analysis in the turbulent cases. Here, we provide a detailed description of the numerical algorithms used in the direct numerical simulation in the analysis of the turbulent flows.

In the turbulent cases, the state system is the high Reynolds number transient Navier-Stokes equation. Adjoint and incremental systems are all linearized Navier-Stokes-structured equations. Thus their treatments are very similar to that of the Navier-Stokes equations. Moreover, the state and the incremental state equations are both forward in time and they are solved conveniently at the same time. The adjoint and incremental adjoint equations are backward in time and thus also solved together given the solutions from the forward systems. Please refer to Appendix B for the details in solving the two backward systems. Here, we focus on the numerical algorithms for solving the state equations and highlight the special treatment for the incremental state equations.

The state equations, the Navier-Stokes equation for an incompressible fluid,

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla p + \frac{1}{Re}\nabla^2\mathbf{u} - \mathbf{u}\cdot\nabla\mathbf{u}, \tag{A.1}$$

$$\nabla\cdot\mathbf{u} = 0, \tag{A.2}$$

are to be solved in a rectangular domain $[0, L_x] \times [-1, 1] \times [0, L_z]$ with periodic boundary conditions for $\mathbf{u}$, p and traction $\tau = [-Ip + \nu(\nabla\mathbf{u} + \nabla\mathbf{u}^T)]\mathbf{n}$ in x and z, i.e,

$$\mathbf{u}(x,y,z,t) = \mathbf{u}(x+L_x,y,z,t) = \mathbf{u}(x,y,z+L_z,t),$$

$$p(x,y,z,t) = p(x+L_x,y,z,t) = p(x,y,z+L_z,t),$$

$$\tau(x,y,z,t) = \tau(x+L_x,y,z,t) = \tau(x,y,z+L_z,t),$$

and

$$\mathbf{u}(y=\pm 1) = 0.$$

There is an uniform pressure gradient so that:

$$\nabla p = \nabla p' + \mathbf{G}, \tag{A.3}$$

where p' is periodic in x and z and $\mathbf{G} = \pi i$ is a vector in x direction. Here, *Reynolds number* $Re = \frac{UL}{\nu}$ is a dimensionless variable with ν denoting the *kinematic viscosity*.

To solve the Navier-Stokes equation for $\mathbf{u}$, we follow the approach proposed by Kim *et al* [40], which eliminates the pressure term from the equations and automatically satisfied the continuity equation. That is, we use a Fourier Galerkin method in x and z directions and a Legendra Galerkin Discretization in y direction, a low storage IMEX scheme (described in Chapter 6) for the time integration.

With the identity that

$$\mathbf{u}\cdot\nabla\mathbf{u} = \nabla(\frac{\mathbf{u}^2}{2}) - \mathbf{u}\times(\nabla\times\mathbf{u}),$$

defining $\omega = \nabla\times\mathbf{u}$, $\mathbf{H} = \mathbf{u}\times\omega$, (A.1) is equivalent to:

$$\frac{\partial\mathbf{u}}{\partial t} = -\nabla p' + \frac{1}{Re}\nabla^2\mathbf{u} - \nabla(\frac{\mathbf{u}^2}{2}) + \mathbf{H} - \mathbf{G}. \tag{A.4}$$

To satisfy the periodic boundary conditions in the x and z directions, $\mathbf{u}$ is represented as a truncated Fourier series in x and z,

$$\mathbf{u}(x,y,z,t) = \sum_{k_x}\sum_{k_z}\hat{\mathbf{u}}(k_x,k_z,y,t)e^{ik_xx}e^{ik_zz}, \quad (\text{where } i=\sqrt{-1}) \tag{A.5}$$

where with given integers N_x, N_z, k_x, k_z denote the wave numbers defined by:

$$
\begin{aligned}
k_x &= \frac{2\pi j}{L_x}, \qquad j = -(\frac{N_x}{2}-1), -(\frac{N_x}{2}-2), \cdots, -1, 0, 1, \frac{N_x}{2}-2, \frac{N_x}{2}-1, \\
k_z &= \frac{2\pi m}{L_z}, \qquad m = -(\frac{N_z}{2}-1), -(\frac{N_z}{2}-2), \cdots, -1, 0, 1, \frac{N_z}{2}-2, \frac{N_z}{2}-1,
\end{aligned}
$$

and $\hat{\mathbf{u}}(k_x, k_z, y, t) = (\hat{u}_x(k_x, k_z, y, t), \hat{u}_y(k_x, k_z, y, t), \hat{u}_z(k_x, k_z, y, t))$ is the Fourier coefficients corresponding to the wave number pair $\hat{\mathbf{k}} = (k_x, k_z)$. Analogously, we assume:

$$
\begin{aligned}
\omega(x, y, z, t) &= \sum_{k_x} \sum_{k_z} \hat{\omega}(k_x, k_z, y, t) e^{ik_x x} e^{ik_z z}, \\
\mathbf{H}(x, y, z, t) &= \sum_{k_x} \sum_{k_z} \hat{\mathbf{H}}(k_x, k_z, y, t) e^{ik_x x} e^{ik_z z},
\end{aligned}
\tag{A.6}
$$

where $\hat{\mathbf{H}}$ is computed from $\hat{\mathbf{u}}$, as we will discuss later.

A.1 Discretization in x and z

Now with the Fourier representation, continuity implies:

$$
ik_x \hat{u}_x + \frac{\partial \hat{u}_y}{\partial y} + ik_z \hat{u}_z = 0, \tag{A.7}
$$

which means that for each wavenumber pair, only two of the velocity components are independent. The cases $(k_x, k_z) \neq (0, 0)$ and $(k_x, k_z) = (0, 0)$ are discussed below.

A.1.1 $(k_x, k_z) \neq (0, 0)$

For this case,

$$
\frac{\partial \hat{u}_y}{\partial y} = -ik_x \hat{u}_x - ik_z \hat{u}_z,
$$

Define

$$\hat{v} = \hat{u}_y,$$
$$\hat{f} = ik_x\hat{u}_x + ik_z\hat{u}_z, \tag{A.8}$$
$$\hat{g} = ik_z\hat{u}_x - ik_x\hat{u}_z.$$

Then (A.7) gives

$$\hat{f} + \frac{\partial \hat{v}}{\partial y} = 0 \tag{A.9}$$

thus $\hat{v}$ and $\hat{g}$ are the only independent variables. To derive the equations for the newly defined variables, we take curl of (A.4) and curl of the curl of (A.4) and conclude:

$$\frac{\partial \boldsymbol{\omega}}{\partial t} = \frac{1}{Re}\nabla^2\boldsymbol{\omega} + \nabla \times \mathbf{H}, \tag{A.10}$$

$$\frac{\partial(\nabla^2\mathbf{u})}{\partial t} = \frac{1}{Re}\nabla^2(\nabla^2\mathbf{u}) - \nabla \times (\nabla \times \mathbf{H}). \tag{A.11}$$

Their second components give a system for $\hat{g}$ and $\hat{v}$,

$$\begin{cases} \frac{\partial \hat{g}}{\partial t} & = \frac{1}{Re}(\hat{\nabla}^2)\hat{g} + \hat{h}_g, \\ \frac{\partial}{\partial t}(\hat{\nabla}^2\hat{v}) & = \frac{1}{Re}(\hat{\nabla}^4)\hat{v} + \hat{h}_v, \end{cases} \tag{A.12}$$

where

$$\begin{cases} \hat{\nabla}^2 & = \frac{\partial^2}{\partial y^2} - k^2, \qquad \hat{\nabla}^4 = \frac{\partial^4}{\partial y^4} - 2k^2\frac{\partial^2}{\partial y^2} + k^4, \\ \hat{h}_g & = ik_z\hat{H}_x - ik_x\hat{H}_z, \qquad k^2 = k_x^2 + k_z^2, \\ \hat{h}_v & = -\frac{\partial}{\partial y}(ik_z\hat{H}_z + ik_x\hat{H}_x) - k^2\hat{H}_y. \end{cases} \tag{A.13}$$

and the boundary conditions:

$$\hat{g}(y = \pm 1) = 0, \quad \hat{v}(y = \pm 1) = 0, \quad \frac{\partial \hat{v}}{\partial t}(y = \pm 1) = 0.$$

A.1.2 $(k_x, k_z) = (0,0)$

For the special case $k_x = k_z = 0$, continuity gives $\frac{\partial \hat{u}_y}{\partial y} = 0$. It implies that $\hat{u}_y = 0$ with the given boundary conditions $u_y(y = \pm 1) = 0$, thus, we only need to solve for $\hat{u}_x(0,0,y,t)$ and $\hat{u}_z(0,0,y,t)$.

Let $U = \hat{u}_x(0,0,y,t)$, $W = \hat{u}_z(0,0,y,t)$, the Fourier coefficients of the first and third components of Navier-Stokes equation (A.1) in $(0,0)$ mode satisfy

$$\begin{cases} \frac{\partial U}{\partial t} & = \frac{1}{Re}\frac{\partial^2 U}{\partial y^2} + \hat{H}_x(0,y,0,t) - \Pi, \\ \frac{\partial W}{\partial t} & = \frac{1}{Re}\frac{\partial^2 W}{\partial y^2} + \hat{H}_z(0,y,0,t), \end{cases} \tag{A.14}$$

and the boundary conditions

$$U(y = \pm 1) = 0, \qquad W(y = \pm 1) = 0.$$

Once we solve the systems (A.12) and (A.14), the Fourier coefficients $\hat{\mathbf{u}} = (\hat{u}_x, \hat{u}_y, \hat{u}_z)$ can be obtained by

$$\begin{aligned} \hat{u}_x &= \frac{k_x \hat{f} + k_z \hat{g}}{i(k_x^2 + k_z^2)} = \frac{k_x \hat{f} + k_z \hat{g}}{ik^2}, \\ \hat{u}_y &= \hat{v}, \\ \hat{u}_z &= \frac{k_z \hat{f} - k_x \hat{g}}{i(k_x^2 + k_z^2)} = \frac{k_z \hat{f} - k_x \hat{g}}{ik^2}. \end{aligned} \tag{A.15}$$

A.2 System Discretization in y

Now we have discretized the Navier-Stokes equation (A.1) in x and z directions. To solve the systems (A.12) and (A.14), we need to further discretize the systems in y. With the boundary conditions, we choose the following discretization using Legendra polynomials:

$$\begin{aligned} \hat{v}(y) &= \sum_{m=0}^{N_y-5} \alpha_m q_m(y), \qquad & \hat{g}(y) &= \sum_{m=0}^{N_y-3} \beta_m \Gamma_m(y), \\ U(y) &= \sum_{m=0}^{N_y-3} a_m \Gamma_m(y), \qquad & W(y) &= \sum_{m=0}^{N_y-3} b_m \Gamma_m(y), \end{aligned} \tag{A.16}$$

where

$$q_m(y) = (1-y^2)^2 P_m(y), \qquad \Gamma_m = (1-y^2) P_m(y), \tag{A.17}$$

and $P_m(y)$ is the m-th order Legendre Polynomial. Note that the boundary conditions in y,

$$\hat{g}(y=\pm 1) = \hat{v}(y=\pm 1) = \frac{\partial \hat{v}}{\partial t}(y=\pm 1) = U(y=\pm 1) = W(y=\pm 1) = 0,$$

now are satisfied by virtue of the choice of basis functions of q_m and Γ_m.

Now plugging (A.16) into systems (A.12) and (A.14), applying Galerkin method, we obtain, for $\hat{v}$, $\hat{g}$,

$$M_v \frac{d\boldsymbol{\alpha}}{dt} = \tfrac{1}{Re} D_v \boldsymbol{\alpha} + \mathbf{F}_V, \tag{A.18}$$

$$M_g \frac{d\boldsymbol{\beta}}{dt} = \tfrac{1}{Re} D_g \boldsymbol{\beta} + \mathbf{F}_g, \tag{A.19}$$

where

$$\begin{aligned} &\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_{N_y-5})^T, \qquad \boldsymbol{\beta} = (\beta_1, \beta_2, \cdots, \beta_{N_y-3})^T, \\ &M_v(m,n) = -\int_{-1}^{1} q'_m q'_n \, dy - k^2 \int_{-1}^{1} q_m q_n \, dy, \qquad M_g(m,n) = \int_{-1}^{1} \Gamma_m \Gamma_n \, dy, \\ &D_v(m,n) = \int_{-1}^{1} q''_m q''_n \, dy + 2k^2 \int_{-1}^{1} q'_m q'_n \, dy + k^4 \int_{-1}^{1} q_m q_n \, dy, \\ &D_g(m,n) = -\int_{-1}^{1} \Gamma'_m \Gamma'_n \, dy - k^2 \int_{-1}^{1} \Gamma_m \Gamma_n \, dy, \\ &\boldsymbol{F}_v(m) = \int_{-1}^{1} q_m \hat{h}_v \, dy, \qquad \boldsymbol{F}_g = \int_{-1}^{1} \Gamma_m \hat{h}_g \, dy. \end{aligned}$$

For U and W,

$$M_0 \frac{d\boldsymbol{a}}{dt} = \tfrac{1}{Re} D_0 \boldsymbol{a} + F_U, \tag{A.20}$$

$$M_0 \frac{d\boldsymbol{b}}{dt} = \tfrac{1}{Re} D_0 \boldsymbol{b} + F_W, \tag{A.21}$$

where

$$M_0(m,n) = \int_{-1}^{1} \Gamma_m \Gamma_n \, dy, \qquad D_0(m,n) = -\int_{-1}^{1} \Gamma'_m \Gamma'_n \, dy,$$

$$F_U(m) = \int_{-1}^{1} \Gamma_m \hat{H}_x^0 \, dy - \Pi \int_{-1}^{1} \Gamma_m \, dy, \qquad F_W(m) = \int_{-1}^{1} \Gamma_m \hat{H}_z^0 \, dy.$$

Note that with the definitions of q_m and Γ_m, M_v, M_g, D_v and D_g are all banded matrices with the j-th superdiagonal and j-th subdiagonal equal to zero if j is odd. In detail, the matrices M_v and D_v have bandwidth 8 (with 9 nonzero diagonals), the matrices M_g, M_0 and D_g have bandwidth 4 (with 5 nonzero diagonals), and the matrix D_0 has bandwidth 2 (with 3 nonzero diagonals).

To solve the systems of ODEs (A.18)-(A.21), we employ the IMEX scheme discussed in Chapter 6, i.e., for (A.18),

$$\begin{aligned} M_v \boldsymbol{\alpha}'_n &= M_v \boldsymbol{\alpha}_n + \Delta t [\frac{1}{Re} D_v (a_1 \boldsymbol{\alpha}_n + b_1 \boldsymbol{\alpha}'_n) + c_1 \boldsymbol{F}_v(\boldsymbol{\alpha}_n)], \\ M_v \boldsymbol{\alpha}''_n &= M_v \boldsymbol{\alpha}'_n + \Delta t [\frac{1}{Re} D_v (a_2 \boldsymbol{\alpha}'_n + b_2 \boldsymbol{\alpha}''_n) + c_2 \boldsymbol{F}_v(\boldsymbol{\alpha}'_n) + d_2 \boldsymbol{F}_v(\boldsymbol{\alpha}_n)], \\ M_v \boldsymbol{\alpha}_{n+1} &= M_v \boldsymbol{\alpha}''_n + \Delta t [\frac{1}{Re} D_v a_3 \boldsymbol{\alpha}''_n + b_3 \boldsymbol{\alpha}_{n+1}) + c_3 \boldsymbol{F}_v(\boldsymbol{\alpha}''_n) + d_3 \boldsymbol{F}_v(\boldsymbol{\alpha}'_n)], \end{aligned} \tag{A.22}$$

where

$$\begin{aligned} &a_1 = \frac{1}{3}, \quad a_2 = -\frac{1}{2}, \quad a_3 = \frac{1}{3}, \quad b_1 = \frac{1}{6}, \quad b_2 = \frac{2}{3}, \quad b_3 = 0, \\ &c_1 = \frac{1}{2}, \qquad c_2 = \frac{1}{3}, \quad c_3 = 1, \quad d_2 = -\frac{1}{6}, \quad d_3 = -\frac{2}{3}. \end{aligned}$$

Similar systems for $\boldsymbol{\beta}$, $\boldsymbol{a}$ and $\boldsymbol{b}$ can be obtained in the same way.

A.3 Pipeline

Now we have finished all the discretization and it remains to figure out how to compute each terms in the linear systems.

The matrices M_g, M_0, D_0, M_v, D_v and D_g in (A.18)-(A.21) do not depend on time and thus can be computed once and reused at each time step. However, there is one of these matrices for each wavenumber pair (k_x, k_z), and it is impossible to store all of them because of the memory restrictions. Nevertheless, each of the integrals in the representations of these matrices can be computed once and stored for reuse since they are independent of the wavenumbers. Furthermore, the nonzero diagonals are the only entries needed to be stored to reduce the need of memory. Since q_m and Γ_m are polynomials and the integrals can be computed preciously using Gauss-Legendre quadrature.

The F terms in the systems are time dependent and thus must be computed at each stages. And the procedure for updating the nonlinear terms in each time step is listed as follows:

- Given $\hat{g}, \hat{v}, U$ and W, determine $\mathbf{\hat{u}}$ by:

$$\begin{aligned}
&\hat{f} = -\frac{\partial \hat{v}}{\partial t} \\
&\hat{u}_z = \begin{cases} \frac{k_x \hat{f} + k_z \hat{g}}{ik^2} & k_x \neq 0 \text{ or } k_z \neq 0, \\ U & k_x = 0 \text{ and } k_z = 0 \end{cases} \\
&\hat{u}_y = \hat{v} \\
&\hat{u}_z = \begin{cases} \frac{k_z \hat{f} - k_x \hat{g}}{ik^2} & k_x \neq 0 \text{ or } k_z \neq 0 \\ W & k_x = 0 \text{ and } k_z = 0 \end{cases}
\end{aligned} \tag{A.23}$$

- Evaluate $\hat{u}_x, \hat{u}_y, \hat{u}_z, \frac{\partial \hat{u}_x}{\partial y}, \frac{\partial \hat{u}_z}{\partial y}$ on a set of $3N_y/2$ Gauss quadrature points y_j, and denote them as $\hat{u}_x^j$ *etc*. Note that applying Galerkin method, the highest order of polynomial to be integrated will be $3N_y$ and thus $3N_y/2$ Gauss quadrature points are needed to get the accurate integration.

- For each triple (k_x, k_z, y_j), compute the components of $\hat{\omega}$:

$$\begin{aligned}
\omega_x^j &= \frac{\partial \hat{u}_z^j}{\partial y} - ik_z\hat{u}_y^j, \\
\omega_y^j &= ik_z\hat{u}_x^j - ik_x\hat{u}_z^j = \hat{g}^j, \\
\omega_z^j &= ik_x\hat{u}_y^j - \frac{\partial \hat{u}_x^j}{\partial y}.
\end{aligned} \tag{A.24}$$

- Use the inverse FFT algorithm to obtain values of $\mathbf{u}$ and ω in physical space. After applying the Galerkin method, the highest wavenumber can be $3N_x$ and $3N_z$. To avoid aliasing error, $3Nx/2$ and $3Nz/2$ points are used for the x and z directions, respectively. And this requires setting the added Fourier coefficients to zero before doing the inverse FFTs. And we conclude the values of $\mathbf{u}$ and ω in a $3N_x/2 \times 3N_y/2 \times 3N_z/2$ grid in physical space.

- At each of the $3N_x/2 \times 3N_y/2 \times 3N_z/2$ grid, compute $\mathbf{H} = \mathbf{u} \times \omega$

$$\begin{aligned}
H_x &= u_y w_z - u_z w_y, \\
H_y &= u_z w_x - u_x w_z, \\
H_z &= u_x w_y - u_y w_x.
\end{aligned} \tag{A.25}$$

- Using FFT to compute the Fourier coefficients $\hat{\mathbf{H}}$ for $\mathbf{H}$.

- Using Gauss quadrature to compute the integrals for Fs.

In real computation, since the solution **u** is real,

$$\hat{\mathbf{u}}(k_x,k_z,y,t) = \hat{\mathbf{u}}^*(-k_x,-k_z,y,t),$$

thus, we need to solve the equations corresponding to the non negative k_x only. So the size of each system is:

- $(N_x/2-1)*(N_z-1)(N_y-4)+(N_z-2)(N_y-4)$ in (A.18);
- $(N_x/2-1)*(N_z-1)(N_y-2)+(N_z-2)(N_y-2)$ in (A.19);
- N_y-2 in (A.20);
- N_y-2 in (A.21).

A.4 Incremental State equations

Now let us talk about the special attention needed for the incremental state system. The incremental state equations are given by:

$$\begin{aligned}
\frac{\partial \delta\mathbf{u}}{\partial t} - \nabla\cdot[-I\delta p + \nu(\nabla\delta\mathbf{u}+\nabla\delta\mathbf{u}^T)] + \mathbf{u}\cdot\nabla\delta\mathbf{u} + \delta\mathbf{u}\cdot\nabla\mathbf{u} &= 0, \\
\nabla\cdot\delta\mathbf{u} &= 0, \\
\delta\mathbf{u} &= 0 \quad \text{at } t=0, \\
\delta\mathbf{u} &= 0 \quad \text{for } y=1, \\
\delta\mathbf{u} + (\nabla\mathbf{u}+\nabla\mathbf{u}^T)\cdot\mathbf{n}\mathbf{W}\cdot\mathbf{n} &= 0 \quad \text{for } y=-1.
\end{aligned} \tag{A.26}$$

As linearized Navier-Stokes equations, (A.26) have the same structure as the state equation (A.1) so they are treated exactly except for a few places.

A.4.1 Boundary conditions

Instead of the homogeneous boundary conditions, the boundary condition at $y = -1$ for the incremental state system, is given by:

$$\delta\mathbf{u} + (\nabla\mathbf{u} + \nabla\mathbf{u}^T) \cdot \mathbf{n}\mathbf{W} \cdot \mathbf{n} = 0,$$

where $\mathbf{W}$ is the variation of the boundary shape. As we described earlier, by setting $\mathbf{W}$ to be the sum of the eigenfunctions we can gain all the information needed to compute all the eigenvalues at once by solving only one incremental state and incremental adjoint equations. Thus, we define

$$\mathbf{W} \cdot \mathbf{n} = \sum_{k_x^2 + k_z^2 \neq 0} e^{ik_x x} e^{ik_z z},$$

where $k_1 = k_2 = 0$ means a shift in the wall normal direction, which is not allowed here because of the fixed volume assumption for the computational domain. Then the boundary condition at Γ_b is given by:

$$\begin{aligned}
&\delta u_x = \left.\frac{\partial u_x}{\partial y}\right|_{y=-1} \Big(\sum_{k\neq 0} e^{ik_x x} e^{ik_z z}\Big) =: \sum_{k_x,k_z} \delta\hat{u}_x^0(k_x,k_z,t) e^{ik_x x} e^{ik_z z}, \\
&\delta u_y = 0, \\
&\delta u_z = \left.\frac{\partial u_z}{\partial y}\right|_{y=-1} \Big(\sum_{k\neq 0} e^{ik_x x} e^{ik_z z}\Big) =: \sum_{k_x,k_z} \delta\hat{u}_z^0(k_x,k_z,t) e^{ik_x x} e^{ik_z z}.
\end{aligned} \tag{A.27}$$

Here, first we need to convert $\mathbf{W}$ from Fourier space to physical space using inverse FFT and then compute the boundary conditions in the physical space. Finally, to solve the incremental state system, FFT will be used again to get the Fourier coefficients of the boundary conditions, which are denoted as $\delta\hat{u}_x^0$ and $\delta\hat{u}_z^0$ as in (A.27)

A.4.2 Methods of lines

Following the same procedure, we can discretize (A.26) in x and z and conclude the same system with now

$$\delta\mathbf{H} = -\mathbf{u}\cdot\nabla\delta\mathbf{u} - \delta\mathbf{u}\cdot\nabla\mathbf{u} = -\nabla(\mathbf{u}\cdot\delta\mathbf{u}) + \mathbf{u}\times\delta\omega + \delta\mathbf{u}\times\omega = \mathbf{u}\times\delta\omega + \delta\mathbf{u}\times\omega$$

and boundary conditions for $\hat{g}$, $\hat{v}$, U and W as:

$$\begin{aligned} &\hat{g}(y=1)=0, \quad \hat{v}(y=1)=0, \quad \frac{\partial\hat{v}}{\partial y}(y=1)=0, \\ &\hat{g}(y=-1)= \quad ik_z\delta\hat{u}_x^0 - ik_x\delta\hat{u}_z^0 := c_1(t), \end{aligned} \tag{A.28}$$

$$\hat{v}(y=-1)=0, \quad \frac{\partial\hat{v}}{\partial y}(y=-1) = -\hat{f} = -ik_x\delta\hat{u}_x^0 - ik_z\delta\hat{u}_z^0 := c_2(t),$$

$$U(y=1)=0, \quad U(y=-1)=\delta\hat{u}_x^0(0,0,t) := c_3(t),$$

$$W(y=1)=0, \quad W(y=-1)=\delta\hat{u}_z^0(0,0,t) := c_4(t)$$

Corresponding, the y-discretization gives

$$\hat{v}(y) = \sum_{m=0}^{N_y-5}\alpha_m q_m(y) + \frac{c_2(t)}{4}(1-y)^2(1+y), \quad \hat{g}(y) = \sum_{m=0}^{N_y-3}\beta_m\Gamma_m(y) + \frac{c_1(t)}{2}(1-y),$$

$$U(y) = \sum_{m=0}^{N_y-3} a_m\Gamma_m(y) + \frac{c_3(t)}{2}(1-y), \qquad W(y) = \sum_{m=0}^{N_y-3} b_m\Gamma_m(y) + \frac{c_4(t)}{2}(1-y),$$

where $q_m(y), \Gamma_m(y)$ are defined as (A.17). The additional term in each expansion is given to meet the boundary conditions. For simplicity, we use the same notation $\hat{g}, \hat{v}, U$ and W in the discretized system for the incremental state equations, however, they should not be confused with these for the state equations.

Applying Galerkin method to (A.26) and IMEX time integration, we can solve the incremental state equations easily and conveniently with the state equations.

Appendix B

Numerical Algorithms for backwards equations

As derived from in Chapter 5, the adjoint and incremental adjoint systems are linearized Navier-Stokes equation which are backward in time with terminal conditions. With the time history of the forward variables, they can be solved together and similarly to the forward systems.

The strong form of the adjoint equations is:

$$\begin{aligned}
&-\frac{\partial \boldsymbol{\lambda}}{\partial t} = \nabla\cdot[-I\gamma + \nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T) + \nu(\nabla\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}^T)] - \nabla\mathbf{u}^T\cdot\boldsymbol{\lambda} + \nabla\boldsymbol{\lambda}\cdot\mathbf{u},\\
&\nabla\cdot\boldsymbol{\lambda} = 0, \qquad\qquad (\text{B.1})\\
&\qquad \boldsymbol{\lambda} = 0 \text{ at } t = T,
\end{aligned}$$

and boundary conditions $\boldsymbol{\lambda} = 0$ on $y = \pm 1$, where $\mathbf{u}$ is the velocity field solved from (A.1). (B.1) is also a Navier-Stokes structured equation with a negative time derivative term and terminal condition, thus it has to be integrated backwards. The numerical algorithm used is the same with that of the forward system while special attention needed for the nonlinear term

$$\mathbf{H} = -\nabla\mathbf{u}^T\cdot\boldsymbol{\lambda} + \nabla\boldsymbol{\lambda}\cdot\mathbf{u}, \tag{B.2}$$

and the time integration scheme.

B.1 Nonlinear term H

For **H** written in the form (B.2), to update it at every time step, we need to do Fourier transformation for all the components of $\nabla\mathbf{u}$ as well as $\nabla\boldsymbol{\lambda}$, which is computational costly. Instead, we will write it in a form that needs minimal Fourier transformations. Using the equality:

$$\begin{aligned}
\nabla\times(\mathbf{u}\times\boldsymbol{\lambda}) &= (\boldsymbol{\lambda}\cdot\nabla)\mathbf{u}-(\mathbf{u}\cdot\nabla)\boldsymbol{\lambda}+\mathbf{u}\nabla\cdot\boldsymbol{\lambda}-\boldsymbol{\lambda}\nabla\cdot\mathbf{u}\\
&= (\boldsymbol{\lambda}\cdot\nabla)\mathbf{u}-(\mathbf{u}\cdot\nabla)\boldsymbol{\lambda};\\
\boldsymbol{\lambda}\times(\nabla\times\mathbf{u}) &= \varepsilon_{mni}\lambda_n[\varepsilon_{ijk}\frac{\partial u_k}{\partial x_j}] = \varepsilon_{imn}\varepsilon_{ijk}\lambda_n\frac{\partial u_k}{\partial x_j}\\
&= (\delta_{mj}\delta_{nk}-\delta_{mk}\delta_{nj})(\lambda_n\frac{\partial u_k}{\partial x_j}) = \lambda_k\frac{\partial u_k}{\partial x_j}-\lambda_j\frac{\partial u_k}{\partial x_j}\\
&= \nabla\mathbf{u}^T\cdot\boldsymbol{\lambda}-(\boldsymbol{\lambda}\cdot\nabla)\mathbf{u}.
\end{aligned}$$

Thus,

$$\mathbf{H} = -\nabla\mathbf{u}^T\cdot\boldsymbol{\lambda}+\nabla\boldsymbol{\lambda}\cdot\mathbf{u} = -\boldsymbol{\lambda}\times(\nabla\times\mathbf{u})-\nabla\times(\mathbf{u}\times\boldsymbol{\lambda}) \tag{B.3}$$

Note that the above form of nonlinear term **H** can also be derived from the form

$$\mathbf{u}\cdot\nabla\mathbf{u} = \nabla(u^2/2)-\mathbf{u}\times(\nabla\times\mathbf{u}),$$

thus, we have the variational form in the Lagrangian function as

$$\mathbf{u}\times(\nabla\times\delta\mathbf{u})\cdot\boldsymbol{\lambda}+\delta\mathbf{u}\times(\nabla\times\mathbf{u})\cdot\boldsymbol{\lambda},$$

where

$$\begin{aligned}
\delta\mathbf{u}\times(\nabla\times\mathbf{u})\cdot\boldsymbol{\lambda} &= (\nabla\times\mathbf{u})\times\boldsymbol{\lambda}\cdot\delta\mathbf{u} = -\boldsymbol{\lambda}\times(\nabla\times\mathbf{u})\cdot\delta\mathbf{u},\\
\mathbf{u}\times(\nabla\times\delta\mathbf{u})\cdot\boldsymbol{\lambda} &= -(\mathbf{u}\times\boldsymbol{\lambda})\cdot(\nabla\times\delta\mathbf{u}) = -\boldsymbol{\lambda}\cdot(\mathbf{u}\cdot\nabla)\delta\mathbf{u}+\mathbf{u}\cdot(\boldsymbol{\lambda}\cdot\nabla)\delta\mathbf{u}\\
&= -\delta\mathbf{u}(-\mathbf{u}\cdot\nabla\boldsymbol{\lambda}+\boldsymbol{\lambda}\cdot\nabla\mathbf{u}) = -\delta\mathbf{u}(\nabla\times(\mathbf{u}\times\boldsymbol{\lambda})),
\end{aligned}$$

with the properties:

$$a\cdot(b\times c) = b\cdot(c\times a) = c\cdot(a\times b),$$

$$(a\times b)\times(\nabla\times c) = b\cdot(a\cdot\nabla)c-a\cdot(b\cdot\nabla)c,$$

we can conclude:

$$\mathbf{H} = -\boldsymbol{\lambda} \times (\nabla \times \mathbf{u}) - \nabla \times (\mathbf{u} \times \boldsymbol{\lambda}).$$

Now with the new form (B.3), in each time step, the procedure to update **H** will be:

- Evaluate $\hat{u}_x, \hat{u}_y, \hat{u}_z, \frac{\partial \hat{u}_x}{\partial y}, \frac{\partial \hat{u}_z}{\partial y}, \hat{\lambda}_x, \hat{\lambda}_y, \hat{\lambda}_z, \frac{\partial \hat{\lambda}_x}{\partial y}, \frac{\partial \hat{\lambda}_z}{\partial y}$ on a set of $3N_y/2$ Gauss quadrature points y_j, and denote them as $\hat{u}_x^j$ *etc*.

- For each triple (k_x, k_z, y_j), compute the components of ω:

$$\begin{aligned}
\omega_x^j &= \frac{\partial \hat{u}_z^j}{\partial y} - ik_z\hat{u}_y^j; \\
\omega_y^j &= ik_z\hat{u}_x^j - ik_x\hat{u}_z^j = \hat{g}^j; \\
\omega_z^j &= ik_x\hat{u}_y^j - \frac{\partial \hat{u}_x^j}{\partial y}.
\end{aligned} \tag{B.4}$$

- Use the inverse FFT algorithm to obtain values of $\mathbf{u}$, $\boldsymbol{\lambda}$ and ω in physical space. $\frac{\partial u_y}{\partial y}, \frac{\partial u_x}{\partial y}, \frac{\partial u_z}{\partial y}$ and $\frac{\partial \lambda_x}{\partial y}, \frac{\partial \lambda_y}{\partial y}, \frac{\partial \lambda_z}{\partial y}$.

- At each of the $3N_x/2 \times 3N_y/2 \times 3N_z/2$ grid, compute $\boldsymbol{\lambda} \times \omega$ and $\mathbf{u} \times \boldsymbol{\lambda}$

$$\begin{aligned}
&(\boldsymbol{\lambda} \times \omega)_x = \lambda_y w_z - \lambda_z w_y =: T_x, \qquad (\mathbf{u} \times \boldsymbol{\lambda})_x = u_y\lambda_z - u_z\lambda_y =: S_x, \\
&(\boldsymbol{\lambda} \times \omega)_y = \lambda_z w_x - \lambda_x w_z =: T_y, \qquad (\mathbf{u} \times \boldsymbol{\lambda})_y = u_z\lambda_x - u_x\lambda_z =: S_y, \\
&(\boldsymbol{\lambda} \times \omega)_z = \lambda_x w_y - \lambda_y w_x =: T_z, \qquad (\mathbf{u} \times \boldsymbol{\lambda})_z = u_x\lambda_y - u_y\lambda_x =: S_z, \\
&\qquad \frac{\partial S_x}{\partial y} = \frac{\partial u_y}{\partial y}\lambda_z + u_y\frac{\partial \lambda_z}{\partial y} - \frac{\partial u_z}{\partial y}\lambda_y - u_z\frac{\partial \lambda_y}{\partial y}, \\
&\qquad \frac{\partial S_z}{\partial y} = \frac{\partial u_x}{\partial y}\lambda_y + u_x\frac{\partial \lambda_y}{\partial y} - \frac{\partial u_y}{\partial y}\lambda_x - u_y\frac{\partial \lambda_x}{\partial y}.
\end{aligned} \tag{B.5}$$

- Using FFT to compute the Fourier coefficients $\hat{T}_x, \hat{T}_y, \hat{T}_z, \hat{S}_x, \hat{S}_y$ and $\hat{S}_z$ $\frac{\partial \hat{S}_x}{\partial y}$ and $\frac{\partial \hat{S}_z}{\partial y}$ for all the components of T and S.

- Compute $\hat{\mathbf{H}}$ by

$$\begin{aligned}
\hat{\mathbf{H}}_x &= -\hat{T}_x - (\nabla \times \hat{S})_x = -\hat{T}_x - (\frac{\partial \hat{S}_z}{\partial y} - ik_z\hat{S}_y), \\
\hat{\mathbf{H}}_y &= -\hat{T}_y - (\nabla \times \hat{S})_y = -\hat{T}_y - (ik_z\hat{S}_x - ik_x\hat{S}_z), \qquad\qquad (B.6) \\
\hat{\mathbf{H}}_z &= -\hat{T}_z - (\nabla \times \hat{S})_z = -\hat{T}_z - (ik_x\hat{S}_y - \frac{\partial \hat{S}_x}{\partial y});
\end{aligned}$$

  Note that the $y-$ derivatives are left for a moment and they will be taken care of in assembling the **F** terms by integrations by parts.

Now consider the incremental adjoint equations:

$$\begin{aligned}
-\frac{\partial \hat{\boldsymbol{\lambda}}}{\partial t} - \nabla \cdot \left(-\mathrm{I}\hat{\alpha} + \nu(\nabla\hat{\mathbf{u}} + \nabla\hat{\mathbf{u}}^T) + \nu(\nabla\hat{\boldsymbol{\lambda}} + \nabla\hat{\boldsymbol{\lambda}}^T)\right) &= \\
-\nabla\hat{\mathbf{u}}^T\boldsymbol{\lambda} - \nabla\mathbf{u}^T\hat{\boldsymbol{\lambda}} + \nabla\boldsymbol{\lambda}\hat{\mathbf{u}} + \nabla\hat{\boldsymbol{\lambda}}\mathbf{u}&, \\
\nabla \cdot \hat{\boldsymbol{\lambda}} &= 0, \qquad\qquad (B.7) \\
\hat{\boldsymbol{\lambda}} &= 0 \qquad \text{at } t = T, \\
\hat{\boldsymbol{\lambda}} + (\nabla\boldsymbol{\lambda} + \nabla\boldsymbol{\lambda}^T)\mathbf{n}(\mathbf{W}\cdot\mathbf{n}) &= 0 \qquad \text{on } \Gamma_b.
\end{aligned}$$

where the nonlinear term is given as

$$\mathbf{H} = -\nabla\hat{\mathbf{u}}^T\boldsymbol{\lambda} - \nabla\mathbf{u}^T\hat{\boldsymbol{\lambda}} + \nabla\boldsymbol{\lambda}\hat{\mathbf{u}} + \nabla\hat{\boldsymbol{\lambda}}\mathbf{u}.$$

Analogously, it can be written equivalently as

$$\mathbf{H} = -\delta\boldsymbol{\lambda} \times (\nabla \times \mathbf{u}) - \nabla \times (\mathbf{u} \times \delta\boldsymbol{\lambda}) - \boldsymbol{\lambda} \times (\nabla \times \delta\mathbf{u}) - \nabla \times (\delta\mathbf{u} \times \boldsymbol{\lambda}).$$

we use the same strategy as above the compute and update it at every time step together with the nonlinear term in the adjoint equations. The boundary condition on Γ_b is treated exactly as described in A.4.1.

B.2 Time integration

To discretize the adjoint equation spatially, we need to choose a gradient-consistent and adjoint-consistent time scheme. Please refer to Chapter 6 for details. The adjoint scheme is in the form of (6.28) with given coefficients given in (6.34).

Additional attention is also needed for the memory issue and please refer to Chapter 5 for the brief discussion of the checkpointing procedure.

Bibliography

[1] Yves Achdou, O. Pironneau, and F. Valentin. Effective boundary conditions for laminar flows over periodic rough boundaries. *Journal of Computational Physics*, 147(1):187–218, 1998.

[2] Robert A. Adams. *Sobolev Spaces*. Academic press, 1975.

[3] V. Akcelik, G. Biros, O. Ghattas, J. Hill, D. Keyes, and B. van Bloeman Waanders. Parallel PDE constrained optimization. In M. Heroux, P. Raghaven, and H. Simon, editors, *Parallel Processing for Scientific Computing*. SIAM, 2006.

[4] A. Almqvist and J. Dasht. The homogenization process of the Reynolds equation describing compressible liquid flow. *Tribology International*, 39(9):994 – 1002, 2006.

[5] A. Almqvist, E. K. Essel, L. E. Persson, and P. Wall. Homogenization of the unstationary incompressible Reynolds equation. *Tribology International*, 40(9):1344 – 1350, 2007.

[6] Andreas Almqvist, Emmanuel Essel, John Fabricius, and Peter Wall. Variational bounds applied to unstationary hydrodynamic lubrication. *International Journal of Engineering Science*, 46(9):891–906, 2008.

[7] Andreas Almqvist, Dag Lukkassen, Annette Meidell, and Peter Wall. New concepts of homogenization applied in rough surface hydrodynamic lubrication. *International Journal of Engineering Science*, 45(1):139–154, 2007.

[8] Youcef Amirat and Olivier Bodart. Numerical approximation of laminar flows over rough walls with sharp asperities. *Journal of Computational and Applied Mathematics*, 164-165:25 – 38, 2004. Proceedings of the 10th International Congress on Computational and Applied Mathematics.

[9] E. Arian. Analysis of the Hessian for aerodynamic optimization. Technical report, Institute for Computer Applications in Science and Engineering (ICASE), 1995.

[10] Eyal Arian and Shlomo Ta'asan. Analysis of the Hessian for aerodynamic optimization: Inviscid flow. Technical report, Institute for Computer Applications in Science and Engineering (ICASE), 1996.

[11] U. Ascher, S. Ruuth, and B. Wetton. Implicit-explicit methods for time-dependent PDE's. *SIAM Journal of Numerical Analysis*, 32:797–823, 1995.

[12] P. R. Bandyopadhyay. Rough-wall turbulent boundary layers in the transition regime. *J. Fluid Mech.*, 180:231–266, 1987.

[13] P. Burattini, S. Leonardi, P. Orlandi, and R. A. Antonia. Comparison between experiments and direct numerical simulations in a channel flow with roughness on one wall. *J. Fluid Mech.*, 600:403–426, APR 10 2008.

[14] J. C. Butcher. *Numerical methods for ordinary differential equations*. Wiley, 2003.

[15] J. Céa. Numerical methods of shape optimal design. In J.Céa E.J. Haug, editor, *Optimation of Distributed Parameter Structures*, volume II, pages 1049–1087. Sijthoff and Noordhoff, Alphen ann den Rijn, Netherlands, 1981.

[16] H. Christensen and K. Tonder. The hydrodynamic lubrication of rough journal bearings. *ASME J. Lubr. Technol.*, 102:368–373, 1980.

[17] J. C. de Alamo, J. Jimenez, P. Zandonade, and R. D. Moser. Scaling of the energy spectra of turbulent channels. *J. Fluid Mech.*, 500:135–144, 2004.

[18] M. C. Delfour and J. P. Zolésio. Velocity method and Lagrangian formulation for the computation of the shape Hessian. *SIAM Journal on Control and Optimization*, 29(6):1414–1442, 1991.

[19] M. C. Delfour and J. P. Zolesio. *Shape and Geometries: Analysis, Differential Calculus and Optimization*. SIAM, 2001.

[20] K. Eppler and H. Harbrecht. A regularized Newton method in electrical impedance tomography using shape Hessian information. *Control and Cybernetics*, 34(1):203–225, 2005.

[21] K. Eppler and H. Harbrecht. Second-order shape optimization using wavelet BEM. *Optimization Methods and Software*, 21(1):135–153, 2006.

[22] K. Eppler, S. Schmidt, V. Schulz, and C. Ilic. Preconditioning the pressure tracking in fluid dynamics by shape Hessian information. *Journal of Optimization Theory and Applications*, 141(3):513–531, 2009.

[23] L. C. Evans. *Partal Differential Equations*. American Mathematical Society, 1998.

[24] A. Griewank and A. Walther. Algorithm 799: revolve: an implementation of checkpointing for the reverse or adjoint mode of computational differentiation. *ACM Trans. Math. Softw.*, 26(1):19–45, 2000.

[25] M. D. Gunzburger. *Perspectives in Flow Control and Optimization*. SIAM, Philadelphia, 2003.

[26] W. W. Hager. Runge-Kutta methods in optimal control and the transformed adjoint system. *Numerische Mathematik*, 87:247–282, 2000.

[27] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration: structure preserving algorithms for ordinary differential equations*. Springer-Verlag, 2002.

[28] H. Harbrecht. Analytical and numerical methods in shape optimization. *Mathematical Methods in the Applied Sciences*, 31(18):2095–2114, 2008.

[29] M. Hintermüller and W. Ring. An inexact Newton-CG-type active contour approach for the minimization of the Mumford-Shah functional. *Journal of Mathematical Imaging and Vision*, 20(1):19–42, 2004.

[30] L. Hormander. Estimates for translation invariant operators in L_p spaces. *Acta Mathematica*, 104(1-2):93–140, 1960.

[31] S. Hoya and J. Jimenez. Scaling of velocity fluctuations in turbulent channels up to $re_\tau = 2000$. *Phys. Fluids*, 18, 2006.

[32] K. Ito and K. Kunisch. *Lagrange Multiplier Approach to Variational Problems and Applications*. SIAM, 2008.

[33] Willi Jäger and Andro Mikelić. On the roughness-induced effective boundary conditions for an incompressible viscous flow. *Journal of Differential Equations*, 170(1):96–122, 2001.

[34] M. Jai. Homogenization and two-scale convergence of the compressible Reynolds lubrication equation modelling the flying characteristics of a rough magnetic head over a rough rigid-disk surface. *Math. Modell. Numer. Anal.*, 29:199–233, 1995.

[35] J. Jiménez. Turbulent flows over rough walls. *Annu. Rev. Fluid Mech.*, 36:173–96, 2004.

[36] J. Jimenez and R. D. Moser. What are we learning from simulating wall turbulence? *Phil. Trans. of the Royal Soc.*, 365:715–732, 2007.

[37] Malal Kane and Benyebka Bou-Said. Comparison of homogenization and direct techniques for the treatment of roughness in incompressible lubrication. *Journal of Tribology*, 126(4):733–737, 2004.

[38] E. Kazantsev. Local lyapunov exponents of the quasi-geostrophic ocean dynamics. *Applied Mathematics and Computation*, 104:217–257, 1999.

[39] E. Kazantsev, J. Sommeria, and J. Verron. Subgrid-scale parameterization by statistical mechanics in a barotropic ocean model. *Journal of Physical Oceanography*, 28:1017–1042, 1998.

[40] J. Kim, P. Moin, and R. Moser. Turbulence statistics in fully developed channel flow at low Reynolds number. *J. Fluid Mech.*, 177:133–166, 1987.

[41] A. Kohl and J. Willebrand. An adjoint method for the assimilation of statistical characteristics into eddy-resolving ocean models. *Tellus Series A*, 54:406–425, 2002.

[42] Armin Kohl and Jurgen Willebrand. Variational assimilation of ssh variability from topex/poseidon and ers1 into an eddy-permitting model of the north atlantic. *Journal of Geophysical research*, 108(C3), 2003.

[43] C. C. Kweh, H. P. Evens, and R. W Snidle. Micro- elastohydrodynamic lubrication of an elliptical contact with transverse and three-dimensional sinusoidal roughness. *ASME J. Tribol.*, 111:577–584, 1989.

[44] Daniel J. Lea, Thomas W. N. Haine, Myles R. Allen, and James A. Hansen. Sensitivity analysis of the climate of a chaotic ocean circulation model. *Quarterly Journal of the Royal Meteorological Society*, 128(586):2587–2605, 2002.

[45] Deniel J. Lea, Myles R. Allen, and Thomas W. N. Haine. Sensitivity analysis of the climate of a chaotic system. *Tellus*, 52A:523–532, 2000.

[46] N. Letalleur, F. Plouraboue, and M. Prat. Average flow model of rough surface lubrication: Flow factors for sinusoidal surfaces. *Journal of Tribology*, 124(3):539–546, 2002.

[47] J. L. Lions. *Optimal control of systems governed by partial differential equations*. Berlin-Heidelberg-New York: Springer-Verlag, 1971.

[48] Vanessa Lopez. *Periodic solutions of chaotic partial differential equations with symmetries*. PhD thesis, University of Illinois at Urbana-Champaign, 2004.

[49] Dag Lukkassen, Annette Meidell, and Peter Wall. Homogenization of some variational problems connected to the theory of lubrication. *Int. J. Engin. Science*, 47:153–162, 2009.

[50] Jean Mathieu and Julian Scott. *An introduction to turbulent flow*. Cambridge University press, 2000.

[51] B. Mohammadi, O. Pironneau, and F. Valentin. Rough boundaries and wall laws. *International Journal for Numerical Methods in Fluids*, 27(14):169–177, 1998.

[52] R. Morrow and P. De Mey. Adjoint assimilation of atlimetric, surface drifter, and hydrographic data in a quasi geostrophic model of the azores current. *Journal of Geophysical research*, 100(25):25,007–25,025, 1995.

[53] R. D. Moser, J. Kim, and N. N. Mansour. Direct numerical simulation of turbulent channel flow up to $re_\tau = 590$. *Phys. of FLuids*, 11:943–945, 1999.

[54] H. M. Nagib and K. A. Chauhan. Variations of von Kármán coefficient in canonical flows. *Phys. of Fluids*, 20:101518, 2008.

[55] J. Nikuradse. Strömungsgesetze in rauhen Rohren. *VDI-Forsch.*, 361, 1933.

[56] L. Pareschi and G. Russo. Implicit-explicit runge-Kutta schemes and applications to hyperbolic systems with relaxation. *Journal of Scientific computing*, 25(1):129–155, 2005.

[57] N. Patir and H. S. Cheng. An average flow model or determining effects of three-dimensional roughness on partial hydrodynamic lubrication. *J. Tribol*, 100:12–17, 1978.

[58] N. Patir and H. S. Cheng. Application of average flow model to lubrication between rough sliding surfaces. *J. Trib.*, 101:220–230, 1979.

[59] S.B. Pope. *Turbulent Flows*. Cambridge University Press, 2000.

[60] M. Prat, F. Plouraboué, and N. Letalleur. Averaged Reynolds equation for flows between rough surfaces in sliding motion. *Transport in Porous Media*, 48(3):291–313, 2002.

[61] Fredrik Sahlin, Andreas Almqvist, Roland Larsson, and Sergei Glavatskih. Rough surface flow factors in full film lubrication based on a homogenization technique. *Tribology International*, 40(7):1025 – 1034, 2007.

[62] A. Sandu. On the properties of rubge-Kutta discrete adjoints. pages 550–557. International Conference for computational Science, Springer-Verlag, 2006.

[63] H. Schlichting. Experimentelle Untersuchungen zum Rauhigkeitsproblem. *Ing. Arch.*, 7:1–34, 1936. Engl. transl. 1937. Experimental investigation of the problem of surface roughness. NACA TM 823.

[64] S. Schmidt and V. Schulz. Impulse response approximations of discrete shape Hessians with application in CFD. *SIAM Journal on Control and Optimization*, 48(4):2562–2580, 2009.

[65] J. Schroter, U. Seiler, and M. Wenzel. Variational assimilation of geosat data into an eddy resolving model of the gulf stream extension area. *Journal of Physical Oceanography*, 23:925–953, 1993.

[66] R. L. Simpson. A generalized correlation of roughness density effects on the turbulent boundary layer. *AIAA J.*, 11:242–244, 1973.

[67] Philippe R. Spalart, Robert D. Moser, and Michael M. Rogers. Spectral methods for the navier-stokes equations with one infinite and two periodic directions. *Journal of Computational Physics*, 96(297 324), 1991.

[68] R. Temam. *Navier Stokes Equations. Theory and Numerical Analysis*. North Holland, Amsterdam, New-York, Oxford, 1979.

[69] Roger Temam. *Navier-Stokes Equations and Nonlinar Functional Analysis*. SIAM, 1983.

[70] A. A. Townsend. *The Structure of Turbulent Shear Flows*. Cambridge Univ. Press, Cambridge, UK, 1976.

[71] R. L. Townsin. The correlation of added drag with surface roughness parameters. In K-S Choi, editor, *Recent Developments in Turbulence Management*, pages 181–191. Kluwer, Amsterdam, 1991.

[72] A. TREVISAN and R. Legnani. Transient error growth and local predictability: a study in the lorenz system. *Tellus A*, 47(1):103–117, 1995.

[73] D. R. Waigh and R. J. Kind. Improved aeroaerodynamic characterization of regular threedimensional roughness. *AIAA J.*, 36:1117–1119, 1998.

[74] P. Wang and T. G. Keith. Combined surface roughness of dynamically loaded journal bearings. *Tribol. Trans.*, 45:1–10, 2002.

[75] C. Wu and L. Zheng. An average Reynolds equation for partial film lubrication with a contact factor. *ASME J. Lubr. Technol.*, 111:188–191, 1989.

[76] J. P. Zolésio. *Identification de domaines par déformation*. PhD thesis, Université de Nice, France, 1979.

Vita

Shan Yang was born in Anhui, China on Aug. 5 1983. She received the Bachelor of Science degree in Scientific Computing from Peking University in July 2005. Then she started the graduate studies in the University of Texas at Austin since Aug. 2005. She got her Master of Science degree in Computational and Applied Mathematics in the Institute of Computational Engineering and Science in May 2007.

Permanent address: 1618 West 6th Street Apt N
Austin, Texas 78703

This dissertation was typeset with LaTeX[†] by the author.

[†] LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.